THE ESSENTIAL ATARI 520ST & 1040ST USER'S GUIDE

How to use and enjoy your ATARI computer, disk drive, printer and peripherals for super computing power!

Includes—

- ☐ Installing & Setting Up
- ☐ Selecting & Using Software
- ☐ Programming With ST BASIC, LOGO and GEM
- ☐ Hardware Interfacing

Jerry Willis

with Gwen Brunson

CAM

WAKE COUNTY PUBLIC LIBRARIES
3 2046 00237 0265

IT'S ESSENTIAL!

book no ATARI ST user can be without. It's the biggest and most comprehensive guide to st and most incredible full-color computers in the world. Discover *everything* you need to know about the amazing ATARI 520ST and 1040ST computers.

With simple, straightforward language, computer expert Jerry Willis shows you how to master your ATARI ST. After learning about setup and installation, you explore the fascinating array of ST software—word processing, educational programs, musical compositions, superb games and more! Enter the fascinating world of telecommunications and other professional applications.

You will also delve into the interesting challenge of programming. Using the GEM operating system, amazing ST BASIC, and LOGO and Turtle graphics for kids of all ages. Expand the possibilities with sound and stimulating graphics. High-level languages like PASCAL and C. All of this and more is possible with ATARI ST computers.

Useful drawings, illustrations, tables, charts and computer printout. Plus, information on ST-compatible accessories and hard-to-find sources of information make this the one-stop Atari ST resource you must have.

About The Author

Jerry Willis is one of the best-selling computer-book authors in the world. He has written more than 30 books on personal computing with sales of nearly 2 million copies. Some of his most popular titles are *The Essential Commodore 128 User's Guide*, *Computers For Everybody*, and *Peanut Butter & Jelly Guide To Computers*. Jerry has a Ph.D. in clinical psychology and lives in Lubbock, Texas.

Here's what reviewers have said about computer books by Jerry Willis:

"(Jerry Willis has) . . . done the masses of computer buyers a great service."
—The New York Times

"Readily understandable to the intelligent adult with no experience. A bargain."
—Library Journal

"Written in a smooth, clear style . . . informative and diverting."
—Datamation

COMPUTER HOW-TO

ISBN 0-89586-511-4

$12.95
$18.50 Canada

KNIGHT-RIDDER PRESS
A Division of HPBooks, Inc.
Tucson, AZ

THE ESSENTIAL ATARI 520ST & 1040ST USER'S GUIDE

Jerry Willis with Gwen Brunson
Senior Editor, Sheryn Johnston

1. Personal Computing and the Atari ST 1
2. A Tour of the ST 9
3. Exploring the ST Desktop 22
4. Exploring the GEM Desktop 40
5. Word Processing on the Atari ST 63
6. Business and Professional Applications 72
7. Telecommunications: A Window on the World 83
8. Home, Recreational, and Educational Applications 104
9. Sound and Graphics on the ST 119
10. Programming in Basic 137
11. More Basic, Sound and Graphics 155
12. Logo and Turtle Graphics 168
13. Machine and Assembly Language 187
14. Other High-Level Languages for the ST 196
15. Accessories and I/O for the ST 207
16. Sources of Information and Software 219
Appendix: Sources of Software and Accessories 225
Index 230

KNIGHT-RIDDER PRESS
Tucson, AZ

WAKE COUNTY PUBLIC LIBRARIES
104 FAYETTEVILLE STREET
RALEIGH, NC 27601

© 1987 HPBooks, Inc.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without prior written permission of the publisher.

The information contained in this book is true and complete to the best of our knowledge. All recommendations are made without any guarantees on the part of the authors and publisher. The authors and publisher disclaim all liability in connection with the use of this information.

Published by Knight-Ridder Press,
A division of HPBooks, Inc.
P.O. Box 5367
Tucson, AZ 85703

Printed in U.S.A.
1st Printing
10 9 8 7 6 5 4 3 2

Library of Congress Cataloging in Publication Data

Willis, Jerry.
The essential Atari 520ST & 1040ST user's guide.

Includes index.
1. Atari 520ST (Computer) 2. Atari 1040ST (Computer)
I. Brunson, Gwen. II. Johnston, Sheryn. III. Title.
IV. Title: Essential Atari 520ST and 1040ST user's guide.
V. Title: Atari 520ST & 1040ST user's guide.
QA76.8.A812W55 1986 004.165 86-82233
ISBN 0-89586-511-4

1

Personal Computing and the Atari ST

This book was written for owners and prospective owners of Atari's ST personal computer. The book was written independently and is not sponsored or controlled by Atari or any of the companies that manufacture peripherals or software for the ST.

The first Atari computer in the ST series was the 520 ST. That model has 512K of Random Access Memory or RAM. The letter K is short for 1024 bytes of memory, enough to store 1024 characters. A newer model, the 1040 ST, has 1040K of RAM which is enough to store over a million characters of data! A few years ago, computers with that much memory cost over $4000, but today ST's are available for a fraction of the cost.

We believe the Atari ST is one of the best personal computer buys on the market today. An ST costs much less than an Apple Macintosh computer, yet with some exceptions it offers many of the same features as the Macintosh. The exceptions, however, are in favor of the ST. For example, both the ST and the Macintosh display excellent high resolution graphics, but the ST does it in color. Both computers can output graphics to a printer, but the ST will work with several popular printers such as the Epson FX-80 while the Macintosh requires special software to work with printers other than models made by Apple. With both computers, you input information and instructions through a keyboard or a mouse, but the ST keyboard has features such as ten "function keys", a numeric keypad, and cursor control keys. Those features are not available on the basic Macintosh model, and even the higher-priced Macintosh models do not have all the keyboard features of the Macintosh.

Why are we comparing the Atari ST to an Apple Macintosh when there are hundreds of personal computer models on the market today? It is because both computers represent a revolution in the way inexpensive personal computers are designed. The Macintosh and the ST are more than new models. They represent a new concept in computing. A traditional computer is very stubborn. To use it you must learn to think and to talk (actually to type) in the jargon of computerese. The Macintosh and the ST, by contrast, are much more sociable. They have been taught to think and behave as if they were an electronic version of the top of your desk. The electronic desktop is made up of icons or images displayed on the video display. There are file folders, documents (represented by pages with one corner turned down), a trash can, even a calculator you can bring out and use. All these

get the machine into production. (It was touch and go for a while, however.) Today there are two models of the ST, with more on the drawing board at Atari, and sales of the computer and the hundreds of programs developed for it are very strong. And who is Atari's major competition today? In many ways it is Commodore and its new computer, the Amiga. It is ironic that the Second Computer War may be fought by Atari, which is now run by former Commodorians, and by Commodore, whose flagship computer was developed with Atari money!

WHAT CAN YOU DO WITH THE ST?

Since many computers sold for business uses cost $3000 or more, many people are skeptical about inexpensive computers. They remember the Timex Sinclairs, the Coleco Adams, and other cheap systems that promised more than they delivered. Let us assure you, the ST really is different. To help you get a perspective on the many uses of the ST, this section will present an overview of the major areas of application. Later chapters will cover topics such as programming, selecting software and accessories, and various applications in more detail.

Computer Literacy

Three hundred years ago more than 97% of the adults in the world could not read. Back then you didn't have to be able to read to be a contributing member of society. Reading and writing was a specialized occupation like shoemaking or blacksmithing. Things have certainly changed! Today we consider the ability to read and write an essential skill if a person is to live successfully and independently in a complex society. The public school system tries to help every citizen to become literate, that is, to be able to read and write.

Some futurists believe our society is moving into a new, information intensive era in which the ability to use computers effectively will be an essential skill for most people. Computer usage, like reading and writing, will become an important aspect of many jobs. We may not be completely literate if we are not "computer literate." The ST is an excellent computer for the individual or family that wishes to become computer literate.

Programming in a High-Level Language

Many people want to use their computers for work. Others use the computer for recreational applications. And still others want to learn how to program the computer. Programming involves typing lines of instructions into the computer. Most people begin their programming work with a high-level language. High-level languages use English-like words and generally work with decimal numbers, the type we learned to add, subtract, multiply, and divide in grade school. The most widely used high-level language today is BASIC, and the ST comes with a good version of BASIC as well as another easy-to-learn language, Logo. Chapters 10 and 11 deal with BASIC, and Chapter 12 covers Logo.

BASIC and Logo are two of the most popular languages today, but there are many others. You can also program the ST in Pascal, C, Modula-2, Forth, and a number of other specialized languages. Chapter 14 introduces programming in high-level languages other than BASIC and Logo.

Machine Language and Assembly Language Programming

Personal computers really understand only one language, machine language. Machine language is not a single language, however. Each computer has a microprocessor chip with a special set of instructions built-in. Instructions to the computer in any other language must be translated into the machine language used by the computer's microprocessor chip, also called the Central Processing Unit or CPU. The ST uses the Motorola 68000 microprocessor chip which is the same chip used by the Apple Macintosh and several other computers. The ST can therefore be programmed in 68000 machine language.

Machine language is the "native" language of a computer, but few people actually write programs in it. It is the language computers understand, but is not easy for humans to learn and use. If you are interested in programming, we suggest you begin with a high-level language such as BASIC or Pascal.

In between high-level languages like Pascal and machine languages is something called assembly language. One of the most trying things about machine language programming is that the symbols used for the instructions are numbers, and not even decimal numbers at that. The numbers that represent computer instructions give humans no indication of their meaning. Unless you memorize the numbers and their meaning, programming in machine language can be very tedious. Assembly languages solve that problem. They let you use letters instead of numbers to give the computer instructions.

For example, in machine language the number 06 might tell the computer to load data into the CPU, while the letters LD might accomplish the same thing in assembly language. Using instructions like LD that stand for LOAD makes it easier to write programs in assembly language, but it is still much more difficult than writing programs in Pascal or some other high-level language. Chapter 13 introduces machine and assembly language.

Recreational Computing

The ST is well suited to the task of entertaining. It has outstanding color graphics, a built-in sound effects generator, and connections for both a mouse and joysticks or game paddles. Several of the better computer games for the ST are reviewed in Chapter 8.

Art and Music

A few years ago few serious musicians or artists considered personal computers anything more than toys or business tools. Today that has changed, and computers like the ST are one reason why. The ST has a built-in interface that allows you to connect many different types of musical instruments to it. If you have a keyboard, electronic drums, or some other instrument that is MIDI (Musical Instrument Digital Interface) compatible, you can probably connect the ST to it. Several companies sell MIDI compatible software that lets you compose and edit music on your computer and then play it through a MIDI compatible instrument.

The ST is one of the few computers with a built-in MIDI connector, but the ST has not ignored those who prefer the visual arts. Many programs turn the computer screen into an electronic canvas where you can create or compose color pictures with the electronic equivalents of a pallet, brush, and drafting instruments. Chapter 9 reviews some of the many ST products for art and sound.

Home and Educational Applications

The ST can perform many important jobs around the home and can become an electronic tutor as well. It can help you keep track of tax information, tutor the kids in basic academic areas, balance your checkbook, and do word processing, among other things. Chapter 8 covers home, recreational, and educational applications.

Business and Professional Applications

Color graphics, a music synthesizer, ease-of-use, and connections for a joystick are some of the features that make the ST an ideal home or school computer. It can also do a hard day's work as a business computer, however. It has a good, full-sized keyboard, a huge memory, high capacity disk drives that are also fast, and a mouse input device that makes it easy to use popular business software such as electronic spreadsheets.

ST programs are available for all of the standard financial and record-keeping functions (e.g., accounts payable/receivable, inventory, general ledger). There are several electronic spreadsheets for the ST (one that works much like the famous Lotus 1-2-3 for the IBM PC), and many word processing programs. Chapter 5 covers word processing and desktop publishing. Chapter 6 covers other business and professional applications.

Telecommunications

A growing number of owners are using their computers for telecommunications. Add a small device called a modem to your ST, and you can connect your computer to the telephone line and communicate with some of the thousands of computers that are programmed to provide services to personal computer owners. Sit back in your easy chair, get a comfortable grip on the keyboard and do everything from paying bills and checking the current price of your blue chip securities to reading the weather report for ski resorts in the Swiss Alps.

With a modem you can communicate with community bulletin boards run by local computer clubs or use the services of national "information utilities" such as The Source and CompuServe. The information utilities bring a wide range of information to you. They provide stock market information, the AP or UPI newswire, shopping services, airline and hotel reservation systems, and databases of information on everything from Alabama sports to zoology. Several of the larger information utilities even have special interest groups for Atari ST computer owners. Chapter 7 provides more information on telecommunications.

WHAT TYPE OF COMPUTER USER ARE YOU?

Before moving on to a discussion of the ST's hardware and software, we should discuss the type of computer user you may want to become. Many people think using a computer involves learning to write programs for the machine. That is one thing you can do, but well over 99% of the people who use a personal computer never write a single program. Here are some categories of users:

Program User

Learns how to run several application programs such as word processors, spelling checkers, and electronic spreadsheets. Knows how to switch on the ma-

chine, run the programs, and how to deal with minor problems that might occur. It takes a few days of effort to become a program user.

Informed User

Does all the Program User does, but has also learned more about the equipment and accessories available and what the computer can and cannot do. Informed Users can intelligently select accessories for their system, install them, and select appropriate programs to accomplish a job or have fun. Becoming an Informed User requires a few weeks of part-time effort plus some continuing reading--such as computer magazines in areas of interest--to keep current on new developments. Many of the chapters in this book were written for readers who want to become Informed Users.

Adaptive User

This person can do all that the Informed User can do. He or she has also learned a bit about how to program a computer. Adaptive Users may not write complete programs, but they can modify or adapt many programs other people have written to better meet current or local needs. They also know more about the inner workings of their computer. In addition, many Adaptive Users can perform some routine maintenance and setup tasks--such as require a cable so a new printer operates correctly with the computer. Becoming an Adaptive User requires several months of part-time effort, perhaps spread over a year or more. Adaptive Users generally read quite a few books on personal computing and subscribe to a number of magazines.

Area or Model Expert

Many ST owners fall into this category. They become interested in a particular area of application such as word processing, music synthesis, or graphics. Or, they become devoted ST owners and try to learn everything they can about the computer. These people spend a great deal of time keeping up with what is happening in their own area of interest. Even if you do not want to become an Expert on the ST or an applications area such as graphics, you may want to get acquainted with several of these people because they are valuable sources of information. Becoming an Expert requires a considerable, and continuing, investment of time. Many Area or Model Experts belong to several local or national computer clubs, and they generally subscribe to a number of relevant magazines.

The Hacker

This person is "into computers." He or she can program a computer in at least one language--BASIC, Pascal or 68000 assembly language--and can start with a blank piece of paper or a blank video display screen and produce a complete working program. Becoming a Hacker requires a great deal of time and an emotional attachment to computers that goes beyond viewing them as useful tools. Many Hackers go on to full-time jobs in the computer industry. Several, in fact, have become millionaires because of their skills. Some have written video games that sold hundreds of thousands of copies. Others have developed business programs that became very popular.

Elitists in the personal computer field, particularly those whose training or interests happen to be in computer science or engineering, tend to look down on people who are not programmers or Hackers. Don't be intimidated. It is perfectly acceptable, and honorable, to use a computer for fun or a particular job. You don't

have to fall in love with the thing or spend most of your weekends slouched over a keyboard. Select the level of use and type of involvement that fits your needs. Then go to it! Using a personal computer such as the Atari ST can take as little or as much of your time as you like.

2

A Tour of the ST

Let's take a quick tour around the ST. It is a small, very powerful computer with many advanced features. In fact, it would probably be better to think of the ST as a computer system instead of just a computer. The system has many different components, and each component makes its own special contribution. This chapter will describe each of the components in the system and offer some suggestions about setting up your equipment.

However, consider the material on setup and installation in this chapter and the next as supplemental. The Owner's Guide Atari packs with the ST shows you how to set up your ST system, and the instructions in that guide are generally correct and thorough. In this chapter we explain what each component in the system does, and we make a few alternative setup suggestions that may save you money and time.

THE POWER SUPPLY

Most of the components in the ST computer use 5 volts of direct current (DC) power. Some components, however, require plus 12 volts DC or minus 12 volts DC. If you have a 520 ST, the large black power supply unit for the ST takes 110 volts AC from your wall outlet and converts it to the three DC voltages the computer needs. Using an external power supply reduces the likelihood of one common problem, heat buildup inside the computer. Power supplies are a major source of heat in a computer system, and using an external supply that connects to the computer through a cable eliminates that source of heat. Unfortunately, it also contributes to the cable clutter that often surrounds a 520 ST computer. If you have a 1040 ST there are far fewer power supply cables to connect because this computer has a built-in power supply and one built-in disk drive.

Power Supply Heat

Several other computer manufacturers, including Apple, IBM, and Tandy, place the power supply inside the main computer unit. That approach has both advantages and disadvantages. Power supplies generate heat, and heat is an enemy of the integrated circuits used in computers. Since the power supply for the 520 ST is an external model, the heat it generates will not build up inside the computer and cause problems.

Other components inside the ST still heat up when the computer is switched on, and the problem is more serious on the 1040 ST since the power supply is built-in. The ST's case has slots on the bottom and top that allow air to circulate. The convection cooling provided helps to keep the ST from overheating. However, the flat space where the cooling slots are located is a handy spot for storing disks, manuals, and program documentation. Avoid the temptation to use that area as a bookshelf. Anything on top of the computer will block the flow of air and might cause overheating.

Protect Disks

In addition, many of the circuits inside the ST generate magnetic fields. If you place a disk on the computer, those magnetic fields might erase part of the data stored on the disk. Keep disks away from magnetic fields! That also means you shouldn't store them on top of the power supply, which generates even stronger fields.

Cable Clutter

The primary disadvantage of external power supplies is that they contribute to the general clutter of cables surrounding many computer systems. Since peripherals such as disk drives, printers, and video monitors have their own power cords, and cords that connect them to the system, your desk may look more like a spaghetti factory gone mad than a high-tech working environment. One way to reduce and control cable clutter, by the way, is to buy a "home" for your ST. Several companies manufacture desktop housings made specifically for the ST. These housings generally cost $30 to $45 and are designed to fit over the ST's keyboard unit. There are cubbyholes for two disk drives and a modem, and the top of the housing is flat so you can place a monitor on it top. Computer Palace sells an ST desktop housing for $29.95. Computer Palace carries many ST products, accepts several credit cards, and has a toll free number for orders [1-800-452-8013].

Each peripheral in an ST will have its own on/off switch, something that's easy to forget about. That should not present a problem. We say "should not" because several years ago we advised the proud owner of a new computer that the external power supply could be left plugged into the wall without fear of problems. A week later an internal fault in the power supply created a short that caused a small fire. Since then we advise people to turn off all the components in their system and unplug external power supplies, which stay on all the time, when the system is not in use.

Powerstrips

All that plugging and unplugging is hard on power cords, however, and the on/off switches in some peripherals will eventually wear out if you use them frequently. We suggest an alternative that reduces wear and tear on switches, helps you clean up cord mess, and provides a measure of protection against power line spikes and glitches. Many computer stores sell powerstrips with built-in switches. Radio Shack, for example, has several powerstrip models. All have three to eight outlets for power cords to plug in. The least expensive powerstrips have no switch and should be avoided. Better powerstrips have a switch you can use to turn on all components at once. Plug your disk drive, printer, video monitor, and ST power supply into the powerstrip. Then flip the switch on the powerstrip to turn all the components on. This approach is convenient and neat.

You will still need to use the on/off switch on the ST since it should generally be switched on last if you have a system with disk drives. Radio Shack's six-outlet powerstrip (#61-2619) is $21.95. It has a circuit breaker and a switch. Similar models are available from many other suppliers.

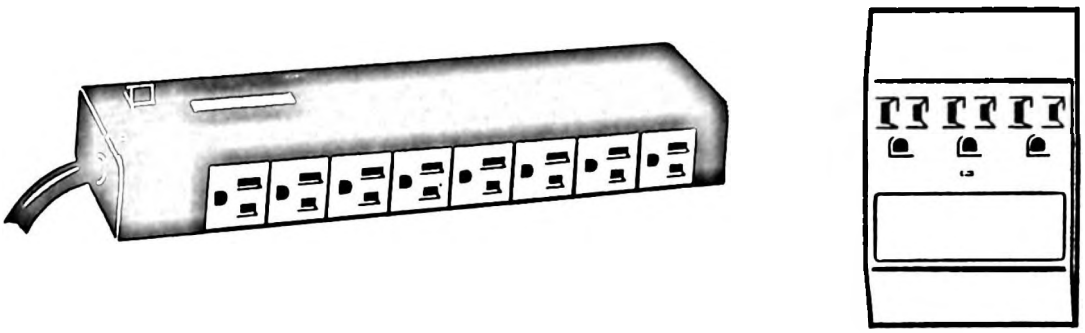


Figure 2.1-A powerstrip with filter and a popular three-outlet filter. (Illustration by Cindy Frisbie)

Power-Line Filters

Powerstrips provide convenience, but they offer little or no protection from power-line glitches and spikes that lurk in virtually every local power system. "Dirty" power can cause your computer to behave erratically, and it can come from many sources. Glitches are generated when the power grid adjusts to changing demands, when large motors in air conditioners or furnaces switch on and off, and when lightning strikes near the power line. To protect your computer from "dirty" power you need a line filter. Radio Shack's power line filter (#26-1451) is also a powerstrip with eight outlets and a switch. At $49.95 it is much more expensive than a powerstrip, but we think the difference in price is a small price to pay for the extra protection it provides. Radio Shack's power filter has only one switch. Other models from companies such as Computer Associates have switches for each of the outlets as well as a master switch that controls all outlets. Models with extra switches and advanced filtering features generally cost $75 to $150.

Some dealers also sell some brands of power filters for under $20. Of the several we have disassembled, none of the so-called filters priced under $20 provided adequate protection.

THE KEYBOARD

The ST's keyboard has 94 keys arranged in four major groups. The main section of the keyboard essentially duplicates the keyboard of a good office typewriter, with a few additional keys. Other sections of the keyboard accomplish different tasks.

Function Keys

The Atari ST has ten programmable function keys across the top of the keyboard. They are labeled F1 through F10. These keys do not have a defined job. Their function is defined by the program controlling the computer. For example, a video game program might use the F1 function key to select the difficulty level of the game. A word processing program might use F5 to indicate the material in the computer's memory should be printed. The shape, design, and feel of the keyboard is generally very good, but the "French cut" of the programmable function keys makes them a bit difficult to use. It is too easy to hit the wrong key. Be especially careful when you strike a programmable function key, especially if hitting the wrong key might be interpreted by the program you are using as an instruction to do something like erase all the information in the memory of the computer.

Numeric Keypad

The ST also has an 18-key numeric keypad for entering large amounts of numeric data. The numeric keypad even has Enter, plus, minus, multiplication (*), and period keys (for typing decimal points). That is a nice touch not found on most computers priced under $1000.

Cursor-control Keys

The four cursor-control keys between the numeric keypad and the rest of the keyboard are one way to move the cursor that is generally displayed on the screen. Striking the key with the right arrow on it moves the cursor -- the marker on the screen that tells you where data entered from the keyboard will appear on the screen -- to the right, and striking the other arrow keys moves the cursor in the direction of the arrow. Cursor keys are used frequently in many programs, but the ST also has another way to move the cursor -- the mouse. It will be discussed in a moment.

Special Function Keys

The ST also has several special function keys that do a particular job. In many programs the Undo key lets you "undo" something you did accidentally. This key is often referred to as the "whoops!" key. The Control and Alternate keys are used to generate special codes. The codes, and the key combinations that generate them, vary from program to program. In general, you will generate a "control code" by holding down the Control key or the Alternate key and pressing another key. For example, some word processors let you print text in boldface by typing Control-B. (Control-B means hold down the Control key and press B. In most programs it does not matter whether you use b or B.) The ESC (short for Escape) key is also used to transmit codes. When you use the ESC key to issue a command, you generally strike it, and then press another key. The ESC key is not held down while you press the second key.

Other keys such as the Help key located near the cursor control keys, and the CLR/Home key (located just above the Right Arrow key) do different things in different programs.

In general, the ST keyboard is the best ever from Atari. It is superior to the keyboards of the IBM PCjr and Apple's IIe and IIc. In fact, we would judge it to be superior to the IBM PC keyboard.

VIDEO DISPLAY

The ST does not have a built-in video display, but it does have the circuits required to connect the computer to a video display. In fact, there are three different video display options for the ST, but all three are not available on all ST models. You can connect the computer to an ordinary black and white or color television, to a special monochrome monitor made by Atari, or to an excellent color monitor made by Atari.

A Regular Television

Some versions of the ST, including the 1040, have a built-in RF modulator that changes the video and sound signals generated by the computer into a signal most televisions can receive. The television thinks the signal is coming from a TV station.

The 520 ST does not have an RF modulator, but if the model you own has one, and if you have a spare television for use with the ST, it will be the least expensive display option. It will also be the least satisfactory. The ST can produce very high resolution graphics, but you will not be able to take full advantage of the ST's video features if you use a television. An ordinary television was not designed to display computer-generated text or graphics. On a TV the display will not be as sharp and crisp as it would be on a special monitor designed specifically for use with the ST.

Monitors

A more satisfactory display for the ST is one of the video monitors made specifically for this computer. Atari sells two types of ST-compatible monitors, the SC1424 which is a monochrome monitor, and the SC1224 which is an RGB color monitor. (The term RGB is explained below.) When you use a television with the ST, data sent to the RF connector on the ST is converted into a radio frequency signal by the computer and then converted back into video and sound signals by the television. The signal loses some of its clarity as it passes through the two conversion processes. That does not happen when the display data is sent to the monitor connector on the ST. The monitor connector uses a round 13-pin plug, and it supplies several types of signals to the monitor. Pin 1 in the connector, for example, contains the audio, or sound, signal. Pin 11 carries the monochrome signal. If you are using the Atari SC1424 monochrome monitor, the signal on Pin 11 carries the data the monitor needs to display an image on the screen. The ST can display text and high resolution graphics on the monochrome monitor.

If you have an SC1224 RGB color monitor, the monitor connector supplies several signals to it. RGB is short for "Red, Green, Blue." RGB monitors are more expensive than either monochrome monitors or lower quality composite color monitors that also display in color, but they do produce sharp, crisp colors. Atari's SC1224 monitor is not only an RGB model, it has special circuits that allow it to produce a display that is even better than that of other RGB models. One Com-

modore magazine summed up the Atari SC1224 by saying "This little RGB analog-only monitor with a single strange Atari-specific input jack produces one of the sharpest pictures of any of the monitors we tested. It is the only reason the Atari's video output looks better than the Amiga's in a side-by-side comparison. Text is sharp, the colors are bright, and it just plain looks good."

Though relatively expensive at around $400, the Atari RGB monitor would seem to be an ideal video display for the ST since it can handle both the graphics produced by the ST and the color signals. There is, unfortunately, a "gotcha." If you purchase the RGB color monitor, you cannot use the high resolution graphics mode. It works only on systems equipped with the monochrome monitor. That means you must buy two different monitors or else choose between color and super-high resolution graphics modes. For many people, color will be more important than the additional resolution. However, if you plan to use the ST primarily for business applications, the monochrome monitor may be a better choice since it offers higher resolution and many of the business programs for the ST do not make extensive use of color. If you have an ST model with an RF modulator built in, you could connect a color television to the system when you play games and use the monochrome monitor when you run business programs.

MEMORY

The 520 ST comes with 512K of RAM or Random Access Memory. The K stands for 1024 bytes of memory, and a byte is enough space to store the code for one character. That means 520K of memory can store 532,480 characters of data. That is a huge amount of memory, especially when 64K of memory was standard even on business computers only a few years ago. The 1040 ST comes with a whopping 1040K of RAM, enough to store more than a million characters of data.

There are a few things you should know about the ST's memory system, however. Many computers load their disk operating systems into RAM. Since DOS takes up some of the available memory, you will not be able to use all the RAM in the computer to load programs or store data. That is not the case with current versions of the ST. Both the 520 and 1040 come with the disk operating system installed permanently in Read Only Memory (ROM) chips inside the computer.

That means the ST has more RAM available for general use than many other models with the same amount of RAM. Note: Early versions of the 520 ST came with the Tramiel Operating System (TOS) on disk. You had to load the operating system into RAM before using the computer. That left very little room for programs and data. All current models have TOS in ROM, and Atari sells an inexpensive upgrade kit to owners of early versions of the 520 ST. If your ST does not have TOS in ROM, we urge you to buy the upgrade kit and install the new ROM chips. That will free up almost half a million bytes of RAM for other uses.

DATA STORAGE OPTIONS

Since everything in RAM is lost when you switch off the computer, a system that will be used for virtually anything other than playing video games must have a more permanent method of storing data and programs. The ST can use three different devices to store data and programs.

The SF354 Disk Drive

Until recently most computers used 5-1/4" disks to store data. A number of newer models, including the ST, are using 3-1/2" disk drives. The Atari SF354 disk drive stores 360K of data on one side of a disk. The drive uses a small disk with a hard plastic case. Inside the disk is a platter of mylar with a magnetic coating. The disk drive "writes" data to the disk by placing magnetic codes on the surface of the disk. When you load programs or data into the computer from a disk, the drive "reads" the magnetic codes.

The great majority of ST owners will probably be satisfied with a system that includes one or two SF354 disk drives. A two-drive system is certainly best because routine tasks such as making a backup copy of a disk are much easier with two drives.

The SF354 stores data on only one side, but it is a high capacity drive that stores as much data on one side of the disk as the 5-1/4" drives for the IBM PC store on both sides of the disk. If you need more storage capacity, Atari has announced the SF314, a high capacity disk drive that stores data on both sides of the disk. The SF314 stores more than a million bytes (one "megabyte") of data on each disk!

If you add disk drives to the ST, can you use drives made for other computers? No, the drives must be specially designed for the ST. Can you buy ST-compatible drives from companies other than Atari? The answer is a qualified yes. By the time you read this, several companies should be offering ST-compatible disk drives that will work with the computer. We urge you to be especially cautious when buying ST-compatible accessories. This is a new model, and several companies have rushed into the market with ads that describe products that are not yet available. For example, a mail order company named Gumball Express placed a full page ad in the June, 1986 issue of *Antic* magazine. The ad described the Shanner SD-2000 disk system for the ST. The SD-2000 costs $399.95 and has two disk drives in the same case. It is the equivalent of two SF314 Atari drives which retail for $299.95 each. The advertisement included a color picture of the Shanner SD-2000 as well as instructions for payment by credit card or by check. The ad, which appeared in a magazine that was on the newsstands in May, said if you paid by check the product "will be shipped when check clears." We called Shanner International, which is based in Mountain View, California, to request an evaluation unit of the SD-2000. The spokeswoman told us they were still working on the drives and, at best, the product would not be available until August of 1986! When you consider ordering a sharp new product from a mail order supplier, always keep in mind that it is much easier to prepare a nice advertisement for a product than it is to design, test, and manufacture that product. A product is not necessarily available just because an advertisement in a magazine says "Available Now."

HARD DISK DRIVES

If your uses for the ST involve storing and retrieving large amounts of data, another storage device, the hard disk drive, may be of interest. Regular, or "floppy," disks can store a few hundred thousand bytes of data. A hard disk drive can store 10 million bytes of data or more, and it operates at 5 to 20 times the speed of regular drives. As you might expect, hard disk drives are expensive. An SF354 drive sells for under $200, while a 10 megabyte hard drive for the ST will cost between $400 and $800.

The relatively high cost of hard disk drives limits their appeal primarily to users with professional and business applications which require more storage capacity than is available on the regular 3-1/2" drives.

Three companies--Atari, Haba, and Supra--already had hard disk drives for the ST when we wrote this chapter. All the hard disks were advertised as compatible with all models of the ST, but many ST owners who purchased these hard disk drives reported problems. In at least some versions of the ST, all the hard disks refuse to run under some circumstances. Apparently the fault is in the ST, not in the hard disk drives. If you try to "boot" the system from the hard disk, the file named DESKTOP.INF produces garbage on the hard disk instead of data.

At the moment you can use any of the hard disks for the ST without problems if you boot from a regular 3 /12" disk and load the DESKTOP.INF file into the computer from that disk instead of from the hard disk. After that, the hard disk should work fine. That is not the way things should work, but you can get up and running with a hard disk by starting from a regular disk.

THE CPU

The CPU, or Central Processing Unit, is the heart of a computer system. Although most CPU's are smaller than a quarter, the electronic components they contain would have filled a room just a few decades ago. Large-scale integration technology permits manufacturers to cram thousands of circuits into tiny silicon chips that work dependably and use less power than a 20-watt light bulb. The Atari ST uses the Motorola 68000 CPU chip, the same one used in the Apple Macintosh and the Commodore Amiga.

At this point you may be wondering whether the ST can run software made for the Macintosh and the Amiga, since all three machines use the same CPU. Unfortunately, compatible CPU's does not mean that Macintosh and Amiga programs will run on the ST. Compatible CPU's make it easier to convert a program from one computer to another, but there is more to compatibility than that. Each computer takes a different approach to things like color and graphics, the way data is stored on the disk, and the way the computer handles programs.

The Atari ST will normally run only programs created specifically for it. We say "normally" because there are ways to run programs written for other computers. For example, at the West Coast Computer Faire in March, 1986, we watched an Atari ST run several programs developed for the Apple Macintosh. That was possible because a company had developed a special circuit card you could add to the ST that made it "Macintosh compatible." Unfortunately, some of the chips needed to build the circuit card are available only from Apple, and that company has shown no interest in supplying those chips to anyone who wants to make the ST Macintosh compatible. When we wrote this chapter, Macintosh compatibility was theoretically possible but practically impossible. That is not the case when it comes to compatibility with two other libraries of software.

CP/M Emulation

The 68000 CPU used in the ST is a very powerful chip. It is so powerful, in fact, that it can run programs that allow it to "emulate" older and slower CPU chips such as the Z80A. That chip was used in many small business computers such as the Kaypro II, Osborne I, and Morrow Micro Decision.

Programs written for computers that use the Z80A chip cannot be run directly on computers that have a 68000 CPU, but there are emulation programs that let you do just that with the ST. For example, Black Patch Systems, a mail order software company sells a CP/M Emulator for the ST. The price is $32.99, and the company has a toll free number for orders (1-800-282-0402). This program al-

lows you to run programs written for computers that use the CP/M operating system. Most of the software created for computers with Z80 and Z80A chips will run under the CP/M operating system.

There are CP/M versions of many famous business programs such as WordStar and SuperCalc that can be run on the ST if you have one of the CP/M emulation programs. Unfortunately, the emulation software is not the only thing you need. Most CP/M programs are supplied on 5-1/4" disks while the ST uses the newer 3-1/2" disks. You must get the CP/M program on a 3-1/2" disk, connect a 5-1/4" disk drive to your ST, or use telecommunications to transfer the program from a CP/M computer to your computer. See the chapter on telecommunications for information on transferring data between two systems.

IBM PC Compatibility

When Atari displayed the ST at the Atlanta COMDEX (short for Computer Dealers Exposition) in May of 1986, it demonstrated two emulation systems. One was for CP/M; the other was for the IBM PC. IBM, and more than 75 other manufacturers, make "PC-compatible" computers that run programs under the MS-DOS or PC-DOS operating system. Most of the computers sold today are PC-compatible, and most new computer programs are available in a version that runs on PC-compatible computers.

The Atari ST has better color graphics, better disk drives, and more built-in features than a standard IBM PC, but there are thousands of programs available for the PC and millions of PC-compatible computers are in use all over the world. Since many of your friends probably use PC-compatible machines, and since there are PC-compatible programs that do everything from draw business charts to keep track of the performance of individual cows in a dairy herd, it would be nice if the ST could run software created for the IBM PC and compatible models. At COMDEX, Atari demonstrated accessory hardware for the ST that lets you run PC software. The product may be available by the time you read this, but we were not able to review it for the book. If it costs less than $400, comes with at least one 5-1/4" disk drive (most PC software comes on 5-1/4" disks), runs most of the software written for the IBM PC, and does not have problems such as slowness, we think this product is worth exploring. That is particularly true if you use IBM PC computers or compatibles at work and have the ST at home.

Help for the CPU

In many computers the CPU chip does all the processing work inside the computer. Data is processed in the CPU, the video display is updated and modified by the CPU, and disk drive operations are controlled by the CPU. The demands placed on the CPU when it must control everything can slow down the system.

That is not a problem with the ST because the 68000 CPU chip has plenty of help. The 68000 does the main computing work, but several other chips are responsible for specific aspects of operation. The YM 2149 chip made by Yamaha, for example, is a specially designed chip that handles sound effects and music synthesis. Other special chips handle the video display tasks, manage the flow of data to and from the computer's memory, and supervise the operation of the disk drives. Atari's extensive use of "intelligent" chips such as the Yamaha sound chip means the ST can perform complex tasks at speeds that would not be possible if the 68000 chip had to handle all of the computing work by itself.

I/O PORTS AND USES

I/O is an abbreviation for Input/Output. If a computer is to be of any use, it must be able to receive information and communicate its response back to you. This basic function is called Input/Output or I/O. The places on the computer circuit board where I/O occurs are often called "ports." The ST has several ports that allow you to communicate with it.

Joystick/Mouse Ports

The keyboard is your primary means of sending information and instructions to the computer, and the port for the keyboard is on the main circuit board. There are also two nine-pin ports on the right side of the ST where joysticks, game paddles, light pens, and the mouse can be connected. Port 1, which is furthest back on the right side, is where you connect the mouse that comes with the computer. Port 0, which is beside the mouse port, accepts a standard joystick.

Atari supplies a two-button mouse with the ST. A computer mouse is a palm sized device you can roll around on your desk. As the mouse is moved over the surface of your table or desk, the movement is electronically coded and sent to the computer. The code received by the computer is generally used to move the cursor around on the screen. For example, if you roll the mouse to the left, the cursor on the screen moves to the left.

Many people like mouse-driven programs. A program may present a menu on the screen and let you roll the mouse around until the cursor is over the option you want to select. Then you press a button on the mouse, and the computer knows what you want to do.

The back and left sides of the ST are also populated with ports. Following is a description of those ports:

Power Connector

The cord from the power supply plugs into the computer through a seven-pin DIN plug on the back of the computer. These round plugs are widely used in Europe, and they are becoming more popular in North America. Most of the connectors on the ST use DIN plugs.

MIDI Out and MIDI In

The two DIN plugs next to the power cord are labeled Midi Out and Midi In. These ports let you connect the ST directly to many different types of musical instruments that are MIDI compatible. MIDI, or Musical Instrument Device Interface, will be discussed in more detail in Chapter 9.

The Monitor Port

Atari's monitors are connected to this port through the Monitor plug. For some reason, Atari elected to use an oddball connector on this port that has thirteen pins in it. No other computer uses this type of plug, but both the monochrome and color monitors made by Atari come with cables that match this connector, so most people will not have a problem because of the weird connector. (If you need a connector that matches the ST monitor plug, you can get one from Alpha Products.)

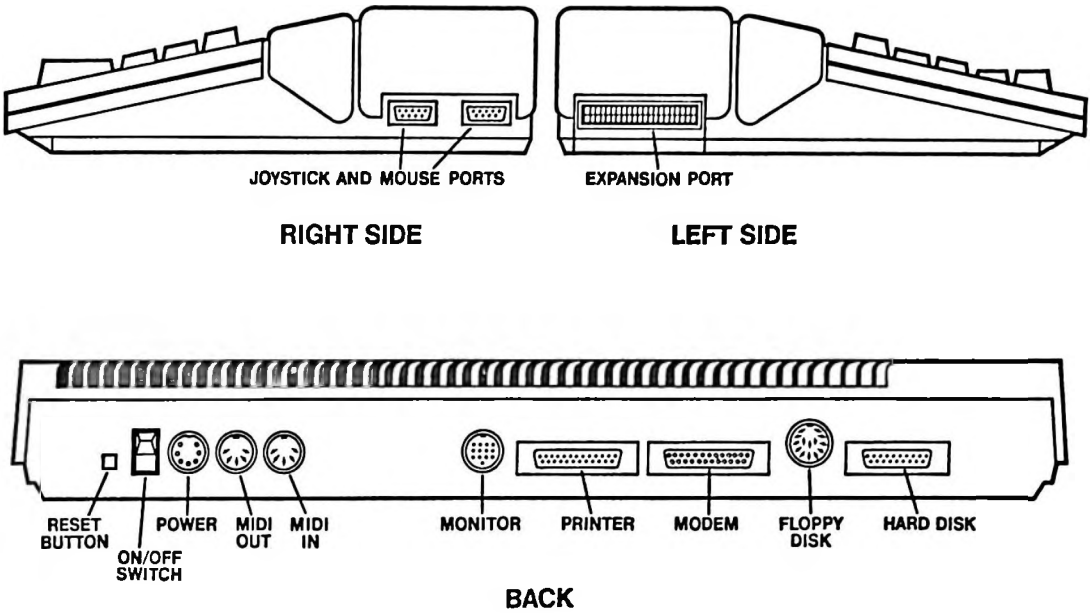


Figure 2.2-Side and Back View of the Atari ST's ports and Connectors. (Illustration by Cindy Frisbie)

Printer Port

Atari seems to have gone out of its way to make it difficult to use a non-Atari monitor with the ST, but it has done just the opposite with the printer port. The ST can use many different types of printers, but most ST owners select the Atari printer or a model from companies such as Epson, Star, or Okidata.

The Printer Port on the ST is actually a general purpose I/O port. There are two types of general-purpose I/O ports in small computers--parallel and serial. The ST's printer port is actually a parallel port that has a wire or connection for each of the eight bits in a byte of data. If you tell the computer to send a byte out a parallel port to a printer, all eight bits in the byte are sent to the printer at the same time over eight different lines. The term Centronics-compatible port refers to the most common type of parallel port. (Centronics is a printer manufacturer that developed this type of parallel port.) The ST's parallel or printer port is Centronics-compatible. If you buy a printer that has a Centronics-compatible parallel interface built-in, you should be able to connect it to the ST through the Printer Port.

The Printer Port uses a 25-pin female D-connector which is not the connector used by Centronics. It is, however, the connector IBM uses on the IBM PC's parallel port. That means you can connect the ST to a parallel printer with the same cable used by the IBM PC. Some stores sell the ST parallel printer cable for as much as $35 because it is a "special item." That special item is often available at another store in town for as little as $10. In the other store it is a popular item, an IBM parallel printer cable. Atari did us a favor by building in a regular parallel port that uses a popular type of cable.

Modem Port

The second type of general purpose I/O port is serial. Serial I/O ports work a bit differently from parallel ports. When eight bits of data arrive at a serial port, they don't all leave at the same time. Instead, the first bit in a byte is sent out on one line, then the next bit goes out on the same line, and so on. All the bits in each byte march out of the computer (or into the computer) over the same line. Any work that involves transmitting data over telephone lines will use a serial port. That is why Atari labels this port the Modem Port. To communicate with another computer over phone lines, you must connect the computer to a device called a modem, and connect the modem to the phone line.

The serial port on the ST will most often be used to connect a modem to the computer (modems are discussed in more detail in the chapter on telecommunications), but some printers also use a serial interface. You can connect a modem to the Modem Port using a standard serial cable with 25-pin D connectors on each end. Connecting a printer to the serial port is a bit more complex. This will be discussed in Chapter 15.

The Disk Drive Port

The Disk Drive Port is adjacent to the Modem Port. You plug the first disk drive in your system to this connector. Cables are supplied with the SF354 single-sided drive and the SF314 double-sided drive that fit this connector. Up to two floppy disk drives can be connected to this port.

The Hard Disk Port

Few computers come with a hard disk drive port built-in, but Atari has one on the back of the computer. It costs more to build in the circuits to handle the high-speed data transfer rates of a hard disk drive, but Atari provides some of those circuits on the main circuit board of the ST. That means hard disk manufacturers don't have to include those circuits in every drive they build for the ST. It also means the method of interfacing hard disk drives to the ST is somewhat standardized. In other computers, you must add a hard disk drive controller board to the computer as well as the hard disk itself. Three manufacturers, Atari, Haba, and Supra, currently sell hard disk drives for the ST. All three companies have 10-megabyte models available now. A "10 meg" hard disk drive can store slightly more than 10 million bytes of data. That is a lot of data, but some people actually need more storage. By the time you read this, some manufacturers should have hard disk drives for the ST that store 20, 30, or even 40 megabytes of data. The average storage capacity of a hard disk on a $6000 IBM PC-AT is only 30 megabytes!

The Cartridge Port

Data and control signals move from one place to another inside the computer over copper traces on the circuit board. There are traces that carry data and traces that handle the "address" codes. Each byte of memory in the computer has its own address, and the address traces carry signals that identify which byte of memory is to be used.

The Cartridge Port on the ST brings the data and address traces out to a connector on the right side of the computer, where external devices such as program cartridges can be connected to the lines. The Cartridge Port on the ST accepts cartridges made for the ST. Some companies are supplying their programs in ROM cartridges that plug into this port. If you plug a video game cartridge into

the Cartridge Port, the Read Only Memory chips inside the cartridge are connected to the data and address lines of the ST through the Cartridge Port. As long as the cartridge is connected, the computer can access and use the program stored in the cartridge.

The ST's Cartridge Port can use ROM cartridges with more than 131,000 bytes of data in them, enough for large business programs as well as sophisticated video games. Thus far, few companies have produced programs for the ST in ROM cartridges. One company, Soft Logik, has an innovative hardware product that comes in a cartridge, a clock card.

The ST does not have a built-in "clock" that keeps track of the time and date even when the computer is switched off. Clock cards generally come with a small battery which keeps the electronic clock operating even when the card cannot get power from the computer. The battery recharges itself when the computer is on.

Clock cards are handy because the disk operating system uses the data they supply to "time-and-date stamp" files. When you create or update a file, the disk operating system includes the date and time in the file data. Then, if you end up with files on several disks with similar or identical names, you can tell which file is the most recent one by the time and date stamp. The disk operating system for the ST allows (but does not require) you to type in the time and date when you boot the system. With the $50 LogiKhron cartridge from Logik Corporation, you don't have to set the time and date each time the computer is switched on. Circuits in the cartridge keep accurate time and supply the time and date to the computer.

Now that you understand a bit about the components that make up your ST computer system, it is time to put the equipment to work.

3

Exploring the ST Desktop

When Charles Rubin reviewed the Apple Macintosh computer in *Personal Computing* magazine, he concluded, "As a product, Macintosh is part hardware, part software, and part philosophy. It's as if Apple had this idea of some super-easy, super-smart Volkscomputer that would make tremendous computing power as easy to use as a telephone, and then figured out from there how to make it."

Apple did indeed figure out how to make a Volkscomputer that was easy to use, but it overlooked one important aspect of the original Volkswagen. The Volkswagen was a people's car because millions of people could afford it. The Macintosh, however, is not an inexpensive computer. We paid almost $3000 for a 128K Macintosh and a printer when the machine was first announced, and today a 512K Macintosh with a monochrome monitor and a printer will still cost almost $2000. For less than $1100 we can buy an Atari 520ST, a high quality color monitor, and a printer. The Atari ST is, in many ways, an affordable Macintosh with color capability.

The Apple Lisa and Apple Macintosh pioneered use of the "electronic desktop" concept for personal computers. The concept was developed at Xerox's Palo Alto Research Center, it was introduced to personal computers by Apple, and Atari made it affordable. In this and the next chapter, we will introduce and explain the philosophy Charles Rubin was so positive about, and show how you can use it in the ST.

SETTING UP YOUR ST

Several of the books in the *Essential Owner's Guide* series devote one or two chapters to the process of setting up the computer. This book doesn't include much information on setting up the ST because Atari has done a very good job of providing set-up instructions in the ST manuals. In addition, setting up the ST is really very simple. You begin by checking to be sure the On/Off switches on the computer, monitor, disk drives, and printer are set to Off. Then attach the mouse to the connector labeled "0" on the right side of the ST. Images of a mouse and a joystick are embossed in the case between Connector 0 and Connector 1. Then you connect the power cord to the plug labeled Power on the back of the ST and plug it into a wall outlet or power strip. The monitor also has a power cord that must be

plugged in. The monitor comes with a cable that connects to the plug labeled Monitor on the back of the ST and to a plug on the back of the monitor.

The power supplies for each disk drive in the system must also be plugged in and connected to the Power plug on the back of the drive. If you have one disk drive, connect the cable that came with the drive to the plug labeled Floppy Disk on the back of the ST. Then connect the other end of the cable to the plug labeled IN on the disk drive. It is possible to connect the cable to the plug marked OUT, but the drive won't work if you do. Be sure to use the IN connector on the drive.

The ST can use a maximum of two floppy disk drives, and if you have a second drive, you must also set up the power supply for it. Connect the cable that came with the second drive to the OUT plug on the first drive in the system, and connect the other end to the IN plug on the second drive. That's it! The ST is ready to work.

If you have a parallel printer, you must connect the cable for it to the plug marked Printer on the back of the ST and to the connector on the printer. There are many different types of printers on the market, and they come with a bewildering variety of connectors. However, most of the parallel printers for the ST can be connected to the computer using a standard "IBM parallel printer cable."

If you have a serial printer you connect it to the Modem port on the back of computer. See the chapter on accessories and I/O for information on the type of cable you will need.

Now you're ready to begin using the ST. This chapter will give you a working understanding of the ST by taking you through some of the steps in operating it. A little later, you will be asked to turn on the ST and insert one of the disks that came with the computer.

THE ST WORKING ENVIRONMENT

One of the nicest things about the ST is that virtually all the programs it can use are designed to work the same way. That is because the computer's operating system has a "shell" called GEM, or Graphics Environment Manager. A shell is a program that handles the interface between a user and the operating system of a computer. Shells are much like menus in a restaurant. A menu in most restaurants is an organized list of commonly requested options. You can say, "I'll take the shrimp dinner" and that tells the server a great deal. It may say, for example, that you want 12 shrimp, a dinner salad, two vegetables, a sauce, and either apple pie or chocolate cake for dessert. That is because the option you selected from the menu defines more than the entree. The GEM shell or "GEM desktop" does the same thing. Using visual images on the screen, GEM allows you to select many of the most commonly used commands and procedures of the operating system by pointing to them.

The basics of GEM are relatively easy to understand, and when you do, you know at least half of what you need to know to use any of several hundred ST programs. That's quite a contrast to the way most personal computers work. With most of them, each piece of software you buy is likely to work very differently than the other programs you use. Often, in fact, programs work in exactly opposite ways. That means past experience can hinder rather than help you in learning a new program. With the ST that rarely happens. Learning one program should make it easier for you to learn others.

THE CONCEPTUAL ELEMENTS OF THE ST

The video display of the GEM shell was designed to resemble your desktop. We'll use *your* desktop rather than ours, since ours is so messy most of the time you can't see the desk for all the articles, books, disks, manuals, and assorted cables that usually occupy it. In fact, when working with the ST, we usually have to move things around a bit to make a smooth spot for the mouse to roll.

If the electronic desktop simulated on ST's screen were a real desktop, you would use physical movements to get a report, a file, or a book. That file of receipts you need to make out your monthly expense report, for example, is lying on the edge of the desk. When you're ready for it, you just reach over and get it.

Now what if the file of receipts is stored magnetically on a disk instead of in a manila file folder? The manila folder might have the label *87 June Expenses* on it. If you've stored the June expense data in an electronic file instead of a manila folder, you could give the electronic file the same name, *87 June Expenses*. When you turn on the ST and put the disk with that file in the disk drive, the screen can display the names of all the files on that disk. That may happen automatically, or you may have to tell the computer to display the names by pressing a switch on the mouse.

Each file is represented on the screen by a tiny image with a name below it. Files that contain data are represented by images of a stack of papers with one corner on the top sheet turned up. One of those files will be your *87 June Expenses* file. You can't reach for this file and pull it off the screen, but there is an electronic equivalent. You can move the mouse around on the desk and, by doing so, move the pointer (a dark arrow) on the screen. When the pointer is over the image of the file folder labeled *87 June Expenses*, you could click the mouse (press the switch on the top of the mouse once) and the ST would select that file. With electronic files, the mouse does the reaching for you!

ST START-UP PROCEDURES

With that introduction, it is time to switch on the computer and begin working with it. Find the 3-1/2" disk labeled *Atari TOS System Disk* (or something similar). In a moment we will insert it into Drive A. Before doing that, however, we should discuss disk safety procedures.

Examine the Atari TOS System Disk or any of the other disks you have. Do you see a small square hole in the side of the disk? There may be a red or black piece of plastic in that square. On one side of the disk, there is only a small square hole. On the other side there is a larger rectangle in which a piece of red or black plastic (your disks may have a different color plastic) can be moved back and forth (with some effort). That piece of plastic is the *write-protect* tab for the disk. The ST can read data from a disk and put that data into its memory. It can also take data in its memory and write magnetic codes which represent that data onto the surface of the disk.

There are some disks that you don't want the computer to write anything onto. They may contain data or programs you cannot easily replace. Therefore, you don't want to accidentally tell the computer to write new data on those disks, and you don't want to accidentally tell the computer to erase anything already on the disk.

That is where the write-protect tab comes in. Turn the disk so you're looking at the top side. The top side is the one with the label and the small square hole. The bottom is the side in which the metal spindle shows in the middle of the

disk, and it has the larger rectangular hole where you can move the plastic write-protect tab back and forth. If the tab is showing through the square hole in the top side, that is, if the square hole is blocked by the tab, that means this disk can be read from and written to by the ST.

We will only be reading from the disk in this chapter, and we have not yet made a backup copy of the of the TOS System Disk, so the disk should be write-protected before inserting it into the disk drive. To do that, turn the disk over and slide the tab up so it isn't covering the square hole. You may have to use something like the tip of a ballpoint pen to get the leverage needed to slide the tab, but most disks can be protected using your fingernail.

Now the disk is write-protected. Even if you accidentally instruct the computer to write something on the disk, that instruction won't be carried out. To unprotect a disk, simply slide the tab back so that it covers the square hole. If your disks have red tabs, think of red as a danger signal that means the programs and data on the disk can be changed if you give the computer the right (or wrong) instruction.

Now it is time to insert the disk into the disk drive. Insert the end with the metal shutter into the disk drive slot. Once the disk is in, a mechanical lever slides that metal shutter back to expose the round platter inside the hard plastic case. Data is stored magnetically on the platter, or disk, of mylar. You can see the platter by sliding the shutter back yourself, but be careful since even minute particles of dust or oil from your fingers can damage the surface of the disk. That is why the shutter stays shut when the disk is outside the disk drive.

The disk should be inserted into Drive A, which is the first drive in your system, the one that is connected directly to the computer. If you have a second drive, that is Drive B and it will be connected to Drive A.

When you insert the disk into the drive, the top side of the disk should be facing up. If you accidentally try to insert the disk upside down, the disk will slide about two thirds of the way in and then stop. It won't go any further, and trying to force it further could damage the disk or the disk drive. If you insert the disk upside down, pull it out and turn it over. The top of the disk is the one with the large label; the bottom is the one with the metal spindle in the middle.

When you insert a disk correctly into the disk drive, it goes all the way into the slot and snaps in place. If for some reason the disk does not seat properly, you can push the ejection lever just below the slot where the disk was inserted, and it will pop out. Then try again. When the disk is properly inserted, it takes very little pressure to push it into the disk drive. If for some reason you meet considerable resistance, don't use brute force to get the disk into the drive. Something is wrong. Determine what the problem is instead of just pushing harder.

When the Atari TOS System disk is in Drive A, turn on the video monitor, the disk drive or drives, and your computer. The ST's On/Off switch is on the back right of the computer. In a moment, the screen lights up and the Busy light on the front of the disk drive turns red. The computer is reading data from the disk drive into the memory of the computer. After a few seconds, the screen will look like the display in Figure 3.1.

Note: The startup instructions we provide here are different from those in the Owner's Manual supplied with early 520 ST's. Those early models did not have the operating system in ROM and were "booted" or started differently. Since all current models of the ST have the operating system in ROM, we assume your ST does too. If it doesn't, we urge you to get a ROM upgrade kit from your ST dealer.

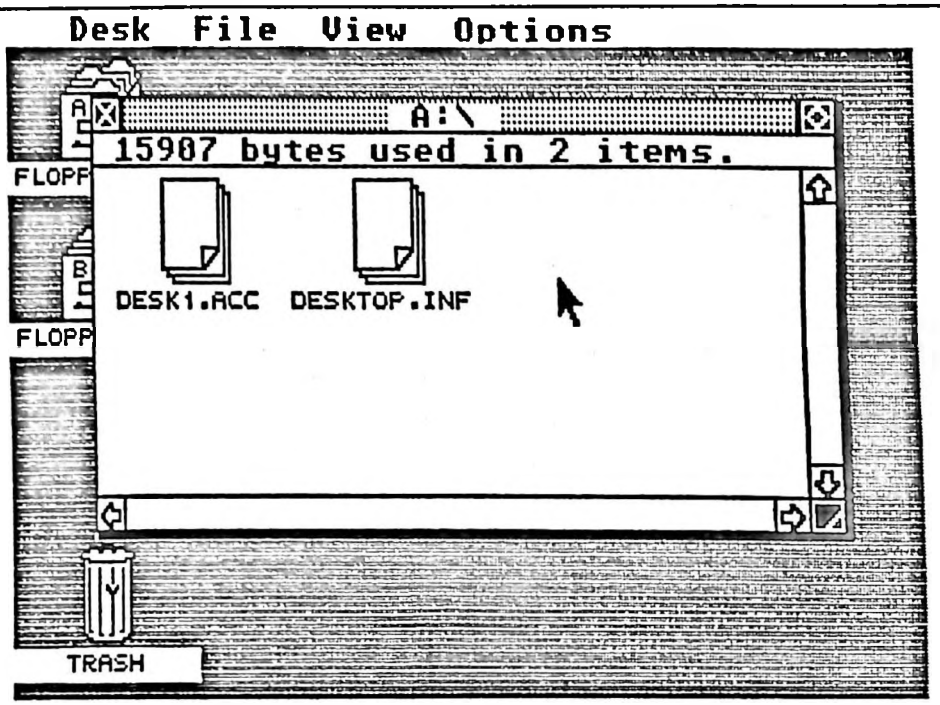


Figure 3.1-This Desktop is displayed when you "boot" the computer using the TOS System Disk.

If the screen seems too dark or too light, the Atari monitors have contrast and brightness controls on the front right under the screen. Use them to adjust the brightness.

The display on the screen is the ST's initial *electronic desktop.* (Our ST will be referred to as *it* in this book, but we recognize that yours may be a male or female model, and may even have a name like Joe, Sue, or Carolyn.)

There should also be a dark arrow on the screen. Make a place for the mouse to roll around on your desk and then move the mouse in various directions. Hold the mouse in the palm of your hand with the cable or "tail" pointing away from you. Note that the arrow on the screen, which we'll call the *pointer* or *screen pointer* from now on, moves in the same direction you move the mouse. With the mouse, you have complete control over the screen pointer. Moving the mouse moves the pointer anywhere on the screen, although it can't be moved outside the boundaries of the screen. Experiment with your mouse a bit to get the idea of mouse movement.

A typical problem is not having enough room to move the mouse so the screen pointer can get from one side of the screen to the other. You may have to move your double-size, no-anchovies deep-dish pizza off the desk to give the mouse operating room. It is not necessary, however, to move the pointer from one side of the screen to the other in one smooth movement of the mouse. For example, if you only have three inches of open space on your desk but need six inches of surface on which to move it, you can drag the mouse across those three inches of open space, pick it up, move it back, and repeat the motion. The effect is the same as rolling the mouse across six inches of open space.

Are you comfortable with the mouse and the screen pointer? If you are, let's explore a little further. The ST comes with several disks. One is labeled *ST Language Disk.* It contains the files needed to prepare the computer to work in the Logo and BASIC computer languages as well as a version of the graphics program

called NEOchrome. Another, labeled *1ST-Word*, contains a very good word processing program that has been customized to work with the ST.

For the examples in this chapter, we will use the disk labeled *Atari TOS System Disk*. It is the disk you inserted in Drive A before turning on the computer. The ST's behavior is controlled by a set of programs called the operating system. Some people say TOS is short for The Operating System, while others say it stands for Tramiel Operating System. In any case, the Atari TOS System Disk has some files on it the computer needs to get ready to work. It loads those files into the memory of the computer when the disk is in Drive A and you switch on the ST.

Unless someone else has used the ST before, there will be only one system disk. Later, you will learn to make copies or backups of any disk, but for now, we have inserted the original disk in Drive A. Generally, it is best to make a copy or backup of any disk that has important data or programs on it, since disks can be damaged by hard use, by accidents such as spilling hot coffee on them, or by a power outage while the disk is in the computer.

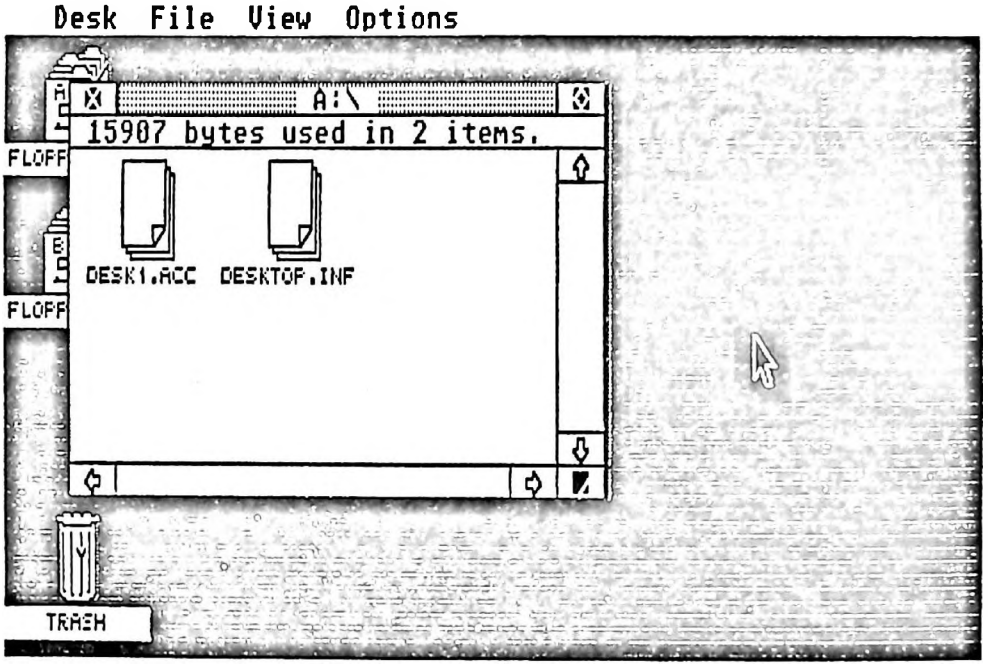


Figure 3.2–This is how the screen looks when the video display uses the "medium" resolution mode.

The first versions of the ST's operating system software had some problems, but most of the problems had been corrected when this book was written. Your ST should work much like the one we describe here, but there is always a chance that changes Atari might make in the hardware or software could result in subtle, or not so subtle, differences. Keep that in mind if your ST does not respond as we indicate it should.

One difference you may notice immediately, if you have a monochrome monitor, is that the illustrations in this chapter and the next do not exactly match your screen. That is because the monochrome monitor can use a higher resolution

display than the color monitor. Our figures show a large trash can, for example, but the one on your display may be much smaller.

WINDOWS AND ICONS

Because the layout and operation of the system disk illustrates the basic concepts of ST computing, we will spend some time exploring how and why things happen with that disk in Drive A.

The screen now has several things on it. On the left side you can see part of the image of two file cabinet drawers. You may not be able to see the labels, but one of the file cabinets is labeled *Floppy Disk A* and the other is labeled *Floppy Disk B*. The file cabinet images are called *icons*. On the same side of the screen at the bottom is another icon labeled *Trash*. As with a real trash can, you can discard things in the trash when you no longer want them.

Most of the screen is taken up by a *window* which is a large rectangle with "A:\" on the top. The A: indicates that the active window on the screen comes from the disk in Drive A. The window for this disk fills up most of the screen.

Let's play with it a bit. A bar of lines across the top of the window has a small square in it on the left side. The square has an X across it. The window is open now, which means it is displaying information. We can close the window by moving the pointer to that square and clicking the mouse. Do that now. Move the pointer (using the mouse) so that it is right over the square in the top left corner of the window. With the tip of the arrow inside that square, click the mouse once. Clicking the mouse simply means pressing the left switch on the front of the mouse once. The window closes and goes back to the file cabinet icon labeled *Floppy Disk A*.

As you move the pointer around near the top of the screen, you may cause one of the menus to drop down. For example, if the pointer comes near the word DESK at the top of the screen, a menu with three options (Desktop Info, Control Panel, and Install Printer) may drop down from the word DESK and take up space on the screen. You don't want to use a menu right now, so move the pointer to a clear area of the screen and click the left button on the mouse. That will remove the menu.

Can you guess how to open the window for Floppy Disk A again? Move the arrow over to the file cabinet icon labeled "Floppy Disk A" and click the mouse once with the pointer on the file cabinet icon. That icon changes from an outline form to a solid form. It is black on a color monitor. When you click the mouse on an icon you *select* that icon. Now move the pointer out to the middle of the screen so it is not on any icon and click the mouse again. That deselects the icon and it changes to the regular outline form.

Selecting the icon did not cause the window to open. To open a window from the icon you must double click the mouse while the pointer is on the icon. Move the pointer over the Floppy Disk Drive A icon and double click it. To double click you quickly press the the left switch on the mouse two times. You must press the left switch quickly to double click, and it may take a few tries to get it done. When you succeed, the screen pointer will change to the shape of a bee to indicate the computer is busy with something. It is reading data for the window it will create from the disk in Floppy Disk A.

Now the window reappears on the screen. Why did you have to double click the mouse to get the window? When the screen has lots of icons on it, requiring you to double click to open a window keeps you from accidentally opening unwanted windows.

Now let's work with the icon labeled Floppy Disk B. Close the window from Disk A by clicking on the square, or *close box* in the top left corner of the window. Then move the pointer over the Floppy Disk B icon. If you have a second disk drive in the system and it is on, place the disk labeled *1st Word* in it. Click the mouse once. The busy light on disk drive B turns red for a moment and a window labeled B:\ appears on the screen.

The line of information at the top of the window says 229534 bytes have been used on the disk and there are 9 items on this disk. Each item is a file that contains either a program or data. For example, the 1st Word program is in a file named *1st Word.PRG*. The icon for that file looks like a hanging file folder or a piece of paper with the top edge folded over. That icon is used for files that contain programs.

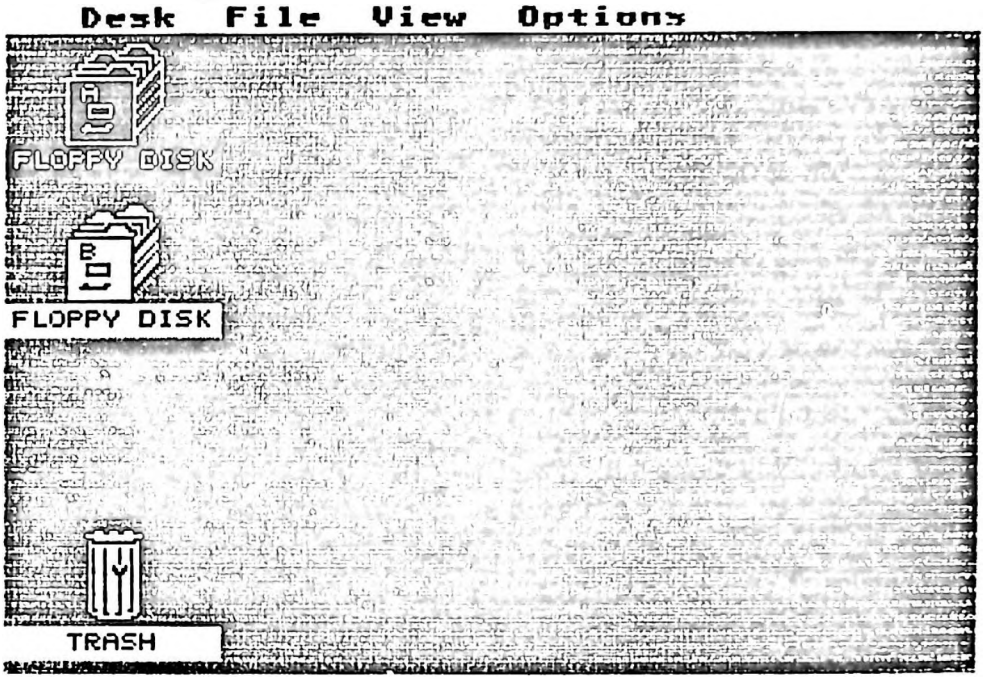


Figure 3.3–The Icon for Drive A has been selected.

Now look at the icon labeled *Guide.DOC*. This file on the disk contains text information. Files that contain data such as text are represented on the screen by an icon that looks like a stack of papers with a corner of the top page turned up.

The third type of file icon is a file folder. File folder icons represent groups of files. Move the screen pointer over the file folder icon labeled *Printer* and double click. That opens a window for the Printer file folder. As you can see, that file folder has several different files in it.

Note that the label at the top of the window says *B:\PRINTER*. That tells you the window you see displays information on the files in the Printer file folder that is on the disk in drive B. The files in this particular folder are used by the 1ST Word program to output text to the particular printer you are using. There are files for many different types of printers.

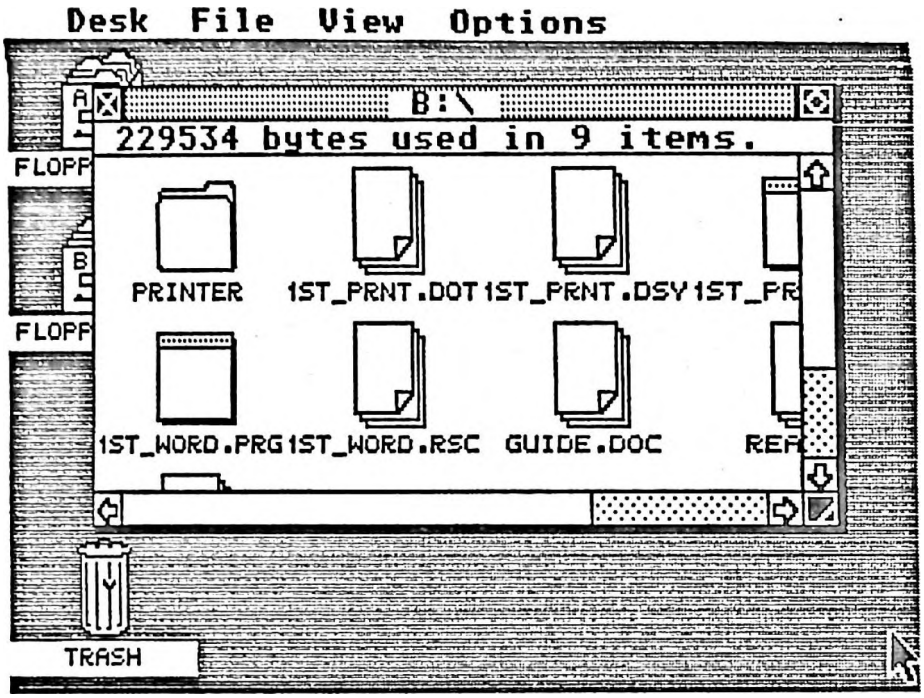


Figure 3.4-The window for the 1st Word disk in Drive B.

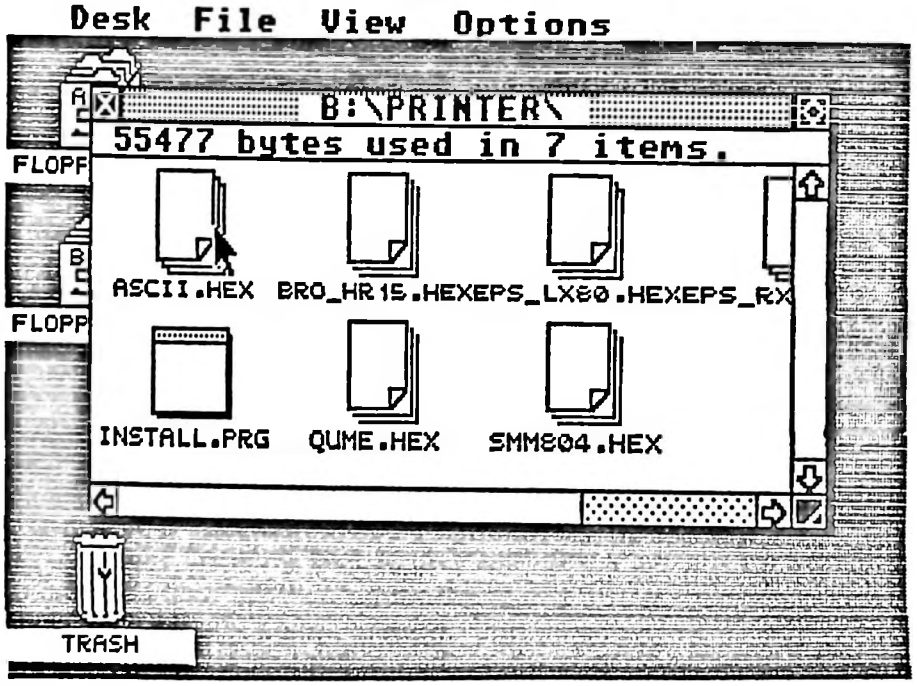


Figure 3.5-The window for the Printer file folder.

As you can see, the ST's GEM desktop is organized much like a small office. There are file cabinet drawers where information is stored (e.g., disks in disk drives), and in those cabinets there are file folders that contain files. Close the Printer file folder by clicking on the close box on the top left of the window. You will be returned to the window for Floppy Disk B. Click on the close box for it, and there will be no open windows on the screen.

You've learned a lot already. Icons are images that represent the electronic equivalents of common items in an office. Icons can be selected by clicking the mouse over them. Double-clicking with the pointer on an icon causes the window associated with that icon to be displayed. Files on a disk can be two types, program and non-program (sometimes referred to as text or document files), and groups of files are represented by file folder icons. And finally, clicking the square in the upper left of a window (called a close box) closes that window.

With that introduction, we move to the menu line at the top of the screen.

MENU BAR

If you still have one or more windows open on the screen, close them by clicking the pointer in the close box of the windows. Now look at the top of the screen. There is a line words: DESK, FILE, VIEW, OPTIONS. This is the menu bar, and each word on that line is called a label. Each of the labels is a menu, or submenu, with its own options.

The menu bar, and the way it works, is one of the most useful and innovative aspects of the ST. Each of the labels on the menu line is the electronic equivalent of a window shade. They remain at the top of the screen until needed. When you need one, you tell it to drop down by moving the screen pointer over the word. Under each label on the menu bar is a menu of options that *drop down* for your use. Note that these are drop-down menus. They drop down whenever the pointer is near the word at the top of the screen. The Apple Macintosh uses pull-down menus. You must move the pointer over the word and then click the button on the mouse to pull a menu down. Some people prefer pull-down menus because they do not drop down when you accidentally move the cursor over a word on the menu bar. Others prefer drop-down menus because they automatically appear when you point to them.

The Options Menu

Many of the commands and features of the operating system can be controlled from the menus that drop down from the menu bar. Other computers require you to learn long lists of cryptic commands and type them to get the computer to work. With drop-down menus you simply select the option you want.

Move the pointer over the label OPTIONS and watch what happens. A drop-down menu appears on the screen. It contains several options. Some options on the menu are in dark, bold print (**Set Preferences, Save Desktop, Print Screen**). Several other options are in light print (Install Disk Drive, Install Application). The options in bold print are available for use now. The ones in light type are not available for use.

Set Preferences

If you haven't done it already, move the pointer down the Options menu. As the pointer passes over each option that is active or available, that option is shown in reverse video (light letters on a dark background). Move your pointer over the

Set Preferences option and then click the left switch on the mouse. To click the mouse, you simply press the left or right switch once. (We will use the term click from now on for this procedure and we will generally refer to the two switches on the mouse as the left and right buttons).

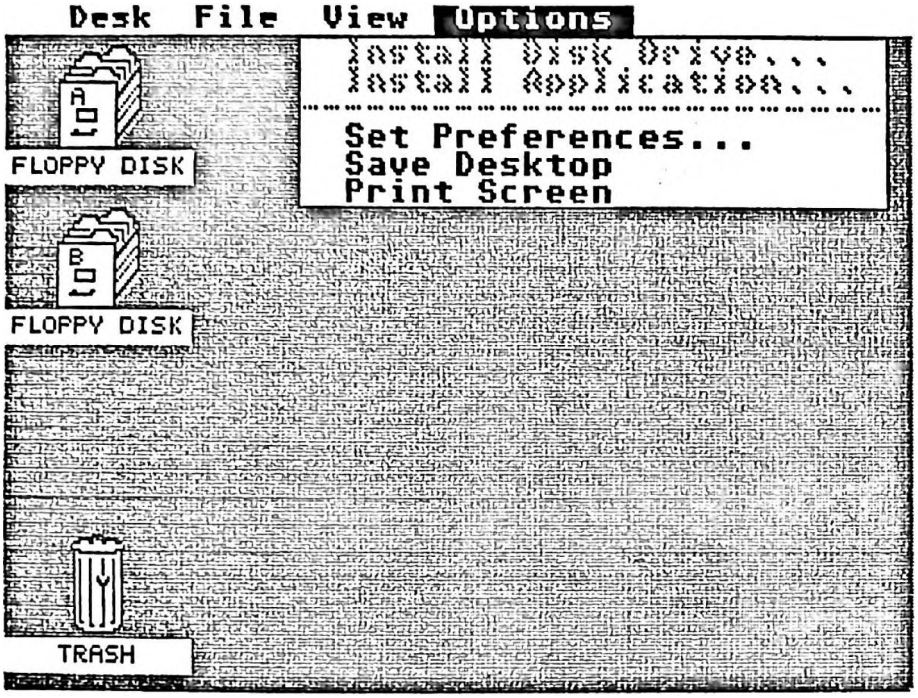


Figure 3.6-The Options drop-down menu.

You have selected the Set Preferences option by clicking the left button while the pointer was over that option. The screen now displays a *dialog box* on the screen. Dialog boxes contain information you need, and most of them list options from which you can select. This dialog box has several options:

Confirm Deletes. When you delete a file by dumping it in the trash can, do you want the computer to ask you to confirm the instruction before it carries it out? The normal or default setting for this option is Yes. That means you will always be asked to confirm an instruction to delete a file before it is done. A dialog box will appear on the screen and list the action the computer plans to take. You click on the OK in the dialog box to confirm that is what you want the ST to do, or you click on Cancel to tell it you don't want it to delete anything.

Move the pointer over the word No beside Confirm Deletes. Click the left button on the mouse and the word No will be in reverse video (light letters on a dark background).

Now you have told the ST you do not want it to check with you before carrying out instructions to delete files. Some people prefer that because it speeds up the process of deleting files. We advise against setting this option to No. When you are in a hurry that is the most likely time to make an error and delete something by mistake. The few seconds of time it takes for the ST to ask for confirmation is worth it.

Move the pointer over the Yes beside Confirm Deletes and click the left button. Now the computer will ask again ask for confirmation before deleting a file or group of files.

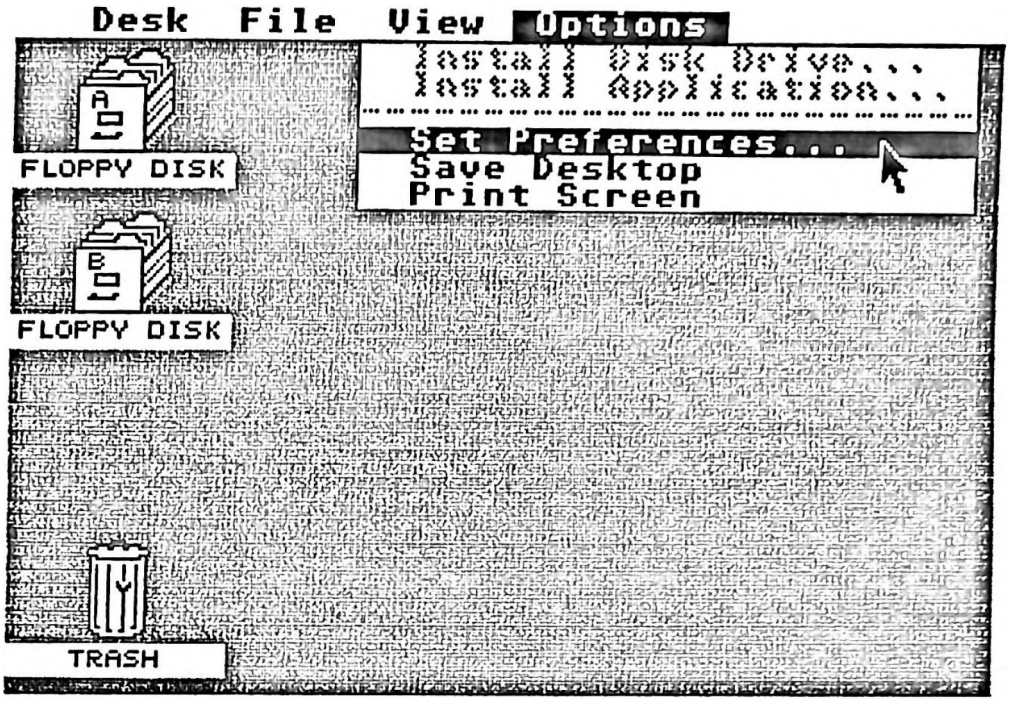


Figure 3.7-The Set Preferences option has been selected.

Confirm Copies. This option works the same way as Confirm Deletes. When you tell the computer to create a copy of a file the computer will confirm the request before beginning work. Again, we think the default setting of Yes is the best, but you can change this option to No if you wish by clicking on the No beside Confirm Copies. Note: The right button on the mouse is rarely used when working on the desktop. From now on when we say "click the mouse" we mean press the left button on the mouse.

Set Screen Resolution. As discussed in Chapter 2, the ST has three different screen resolutions. One of them, the 640 by 400 pixel high resolution monochrome mode is the only one you can use if you have an Atari monochrome monitor. The ST automatically checks to see what type of monitor is connected to it when it is switched on. If you have a monochrome monitor, it will come up in the high resolution monochrome mode, and the medium and low resolution options for Screen Resolution will be faded and barely visible. That is because they are not usable options if you have a monochrome monitor. If you try to select the Low or Medium resolution mode by clicking on them, nothing will happen.

If you have an Atari color monitor, the high resolution mode will not be active. The color monitor can only use the 320 by 200 pixel, 16-color low resolution mode and the 640 by 200 pixel, 4-color medium resolution mode. The default or standard resolution is low, and that option will be shown in reverse video.

Let's change that option to medium resolution. Move the pointer over the Medium and click. That highlights Medium, but nothing else happens. The ST is a cautious machine, and it does not act on options you select in dialog boxes immediately. Instead, it waits until you click on the OK at the bottom of the screen. If you make changes and then decide to forget it, all you have to do is click on Cancel and the dialog box will disappear. This time, move the pointer over OK and click. That

says you are satisfied with the options you have selected and are ready for the computer to act on your instructions.

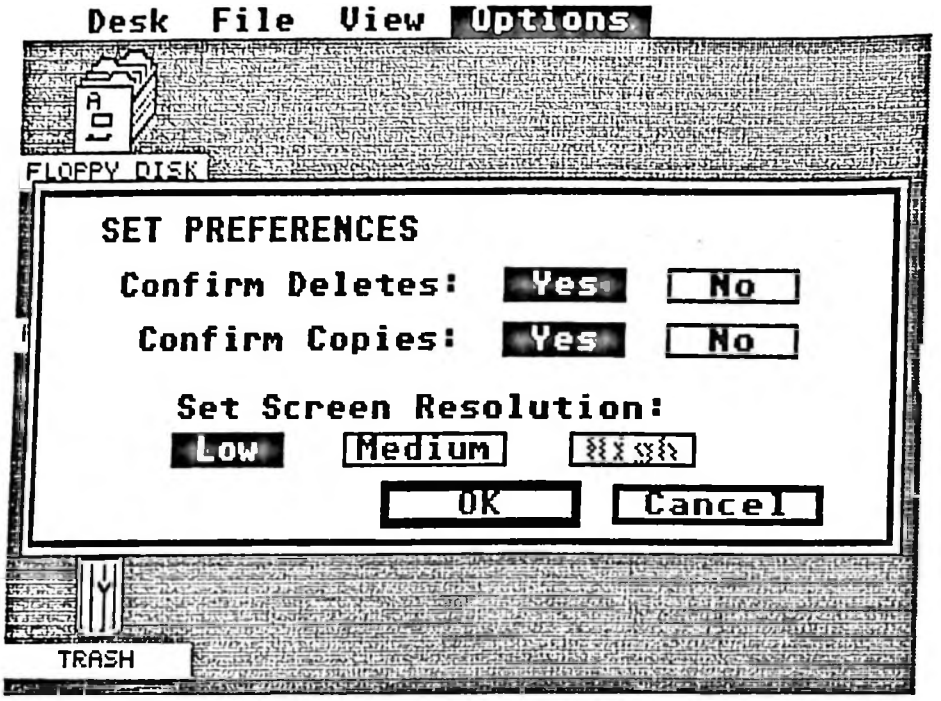


Figure 3.8-The Set Preferences Dialog Box

When you click OK, the dialog box will disappear, and after a moment, the screen will change. Now the icons on the screen are smaller because you are using the medium resolution display mode. In this mode only four colors can be used, but the graphics can be much more detailed.

Since we are using the low resolution mode in all our figures, move the pointer over Options so that the menu will drop down. Then select the Low resolution mode by clicking on Low, and click OK to tell the ST you want to change back to the low resolution mode.

Now your screen will look like the figures in this chapter if you have a color monitor. Close the window that opened for Disk Drive A by clicking on the Close box.

Save Desktop

If you move the pointer over the Options label on the Menu bar, the Options menu will drop down again. Another option is Save Desktop. The GEM desktop has default settings for things like low, medium, or high resolution graphics modes, and for things like whether you are asked to confirm an instruction to delete a file. If you do not like some of the default settings for the desktop, you can change them using options in the drop down menus. However, if all you do is change the default settings, the default options will automatically be used each time you switch on the ST.

To make the preferences you have selected the default settings, select Save Desktop by placing the pointer over Save Desktop on the Options menu and clicking. Your settings will replace the default settings that are in a file on the disk. Now, when you start the computer with that disk in drive A, your settings will be used

instead of the standard defaults. Note, however, that the changes are made only to the disk currently in disk drive A.

Print Screen

If you have a printer and it is properly connected to the computer, you can set a printed copy of the screen display by selecting this option. Is the printer on? Does it have paper, and is it connected to the printer port on the back of the ST? If the answer is yes to all these questions, you can click on Print Screen and watch while hardcopy of the screen is output to the printer. When you select Print Screen, the menu disappears, and the pointer changes to a busy bee. That is the image the printer reproduces.

This method of printing the screen is easy and straightforward, but it is often not what you want. In most cases, for example, you will not want a bee sitting in the middle of the printout. Another way to print the screen is to hold down the Alternate key on the keyboard and press the Help key. That prints the screen as it is. If you do not want the screen pointer to be displayed, move it off the screen. If the pointer is off the bottom right side of the screen, you will not even see the point of it on the printout.

Install Disk Drive

The name of this option is a little misleading. It seems to be something you would do when you buy a new disk drive and add it to your system. You might use this option under those circumstances, but that is not the only time you use it. If you do not have a window open on the screen, open a window for Floppy Disk A. Then point to the Options label at the top of the screen. When the menu drops down, Install Disk Drive is not an active option. It is displayed faintly. Close the window that is open by clicking on the close box. The first time you click on the close box, the Options menu will roll up. The second time you click, the window will close.

Now point to the Options label again. Install Disk Drive is still not active even when there is no window open on the screen. Point to the Floppy Disk A icon on the screen and select it by clicking. If the Options menu is down, the first time you click, the Options menu rolls up. Click again to select Drive A. The icon will change to reverse video.

Now point to Options. Install Disk Drive is now active. Select it by pointing to it and clicking. A dialog window opens up on the screen that has two options on it.

The first option is Drive Identifier; and since you selected Floppy Disk A before using Install Disk Drive, the current drive identifier is A. If you would like to change the identifier, which is printed on the file cabinet icon displayed on the screen, you can do that by pointing at the A beside Drive Identifier in the dialog box, and clicking. Then you can strike the Backspace key on the keyboard to erase the current identifier and type another identifier.

You can do the same thing for Icon Label. The current label is Floppy Disk, but you can erase that with the backspace key and change it to something else if you wish. For example, you could change icon label to "Program Disk" if that suits you.

At the bottom of the dialog box there are three options. If you point to Cancel and click, the dialog box is closed without anything being changed. If you point to Install and click on it, the identifier and icon label displayed in the dialog box will replace the ones currently used by the drive you selected.

The last option at the bottom of the screen is Remove. This is a dangerous option that should generally be avoided. If you point to and click on Remove, the icon specified by the Drive Identifier and Icon label will be removed from the screen. You could, for example, remove the icons for both the A and B drives. That would mean you could to access the disk drives at all. You would have to push the reset button on the back of the computer to boot up from the TOS System disk to "reinstall" the disk drive icons. If you removed both the A and B drive icons, and then used the Save Desktop option, you could not even reinstall the drive icons. You would have to use a backup copy of the TOS Operating System disk that did not have the drive icons removed. In general, we suggest you avoid this option unless there is a specific reason to use it.

You may be thinking that since you have only one disk drive, it would be a good idea to remove the icon for Floppy Drive B. Even if you have only one disk drive, we advise against that. The ST's operating system has a nice feature that lets you work as if you have two disk drives, even if only one real drive is in the system. If a task requires two disk drives, the ST will prompt you to remove the A disk and insert the B disk in the disk drive as needed.

Install Application

This option is a bit difficult to explain because there are several ways it can be used. We will use a theoretical example that illustrates how the Install Application option is used. Don't try to type this example since it uses files that are not on your disks.

Suppose you have written a program in BASIC that is saved in a file name SALESDTA.BAS. You use it to analyze quarterly sales data. Under normal circumstances, using that program would require you to load the BASIC program into the computer first, then you would load and run the SALESDTA.BAS program.

Another way to do it is to use the Install Application option to tell the computer you want to load automatically and start BASIC whenever you select the program in the file named SALESDTA.BAS.

To begin this you would open the window that contains the SALESDTA.BAS program icon. Then you would select the SALESDATA file by clicking on its icon. The next step requires you to pull down the Options menu. The Install Application option will be active and available for use if you selected the icon of a program file before selecting the Options menu.

When you select the Install Application option, a dialog box will be displayed on the screen. The name of the program you selected will be displayed beside the heading Application Name. Below that is the heading, Document Type.

The file name SALESDTA.BAS has two parts. SALESDTA is the name of the program, and BAS is the file extension. The name and the extension are separated by a period. If we type the extension BAS beside the Document Type heading, selecting SALESDTA.BAS will cause the BASIC language program to be loaded into the computer and then the program SALESDTA.BAS will be run. In fact, selecting any program with a BAS extension in the name will cause the BASIC language to be loaded.

If you selected a word processing program before selecting Install Application, typing DOC beside the Document Type heading would tell the ST you want the word processing program loaded when you select a file with DOC as the extension.

At the bottom of the dialog box, there are three options: GEM, TOS, and TOS-takes parameters. Most programs work through GEM and you will click on GEM to indicate that. A few programs will use TOS without going through GEM, and you will need to select either TOS or TOS-takes parameters. Check the docu-

mentation that comes with the applications software if you are unsure of the way the program interacts with the ST's operating system.

Once you have made your choices, you can click on the OK box at the bottom of the dialog box to close the box and confirm the instructions. If you decide to forget the whole thing, click on the Cancel box.

Remember, however, that virtually all the options you select from menus will be active only during the current session. If you switch off the machine, the operating system returns to the default settings. If you want to make your settings the defaults for the operating system, select Save Desktop from the Options Menu.

VIEW

Normally the files in a window are shown as icons with labels below them. That is only one of several ways you can format the screen display. The View menu includes five options:

Show as Icons. This is the normal or default mode as indicated by the check mark beside it. Icons designate various types of files. A letter you wrote in 1st Word is represented by a stack of pages with a corner turned down, a folder of files is represented by a manila folder icon, and programs are represented by a page with the top edge folded over.

Show as Text. This option organizes the files in the window alphabetically by name. This option gives you the name of the file, the type of file (e.g., folder, applications program, document, and so on), the size of the file in K (1 K is 1024 bytes), and the date that file was last modified. You can select this option by clicking on the Show as Text option of the View menu.

Date. The Date option gives the same information as Name, but the files are organized by the last date they were modified beginning with the newest file.

Size. This gives the same data as name and date, but the files are arranged by size. The largest one is listed first. Size is in K.

Kind. This final View option organizes the files by type. Folders are listed first, followed by applications programs, then documents created by programs such as 1st Write and NeoChrome.

THE TRASH CAN

Thus far, the little trash can in the bottom corner of the screen has not been used. We will describe how it works here and provide more details in the next chapter.

In an office, the trash can is where you discard items you no longer want or need. The trash can on the GEM desktop serves the same purpose. You can dump a file or a folder in the trash by pointing to it with the mouse and holding down the left button on the mouse. Then you move the pointer until it is over the trash can. As long as you hold down the left button, an outline of the icon for the file or folder will move with the pointer. This process is called dragging. If you place the pointer over the trash can icon and then release the left button, you have dumped the file or folder that was dragged to the trash.

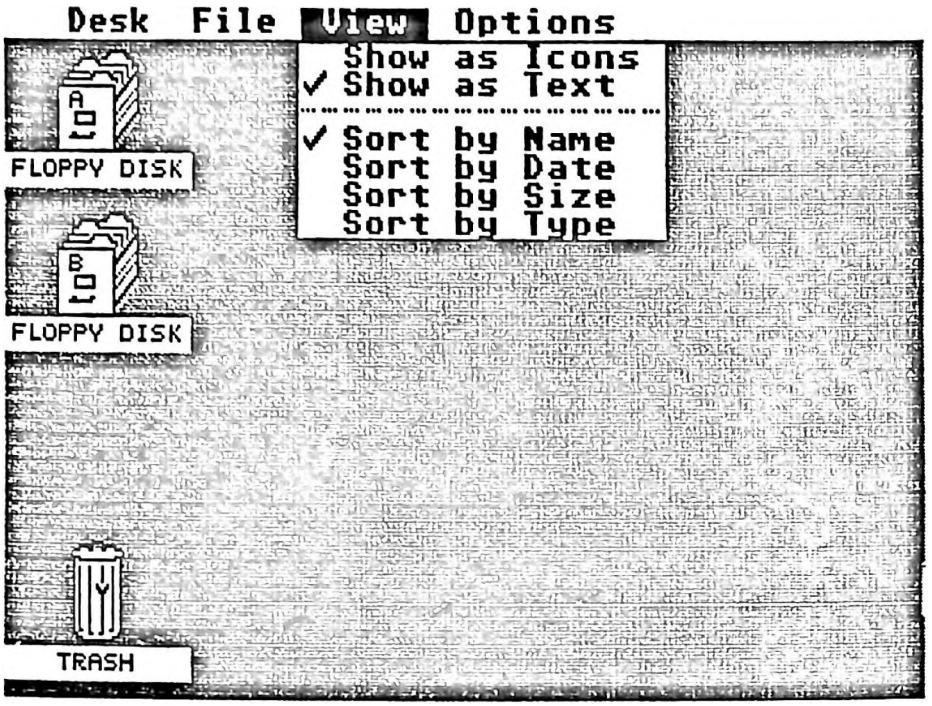


Figure 3.9-The View Option Menu.

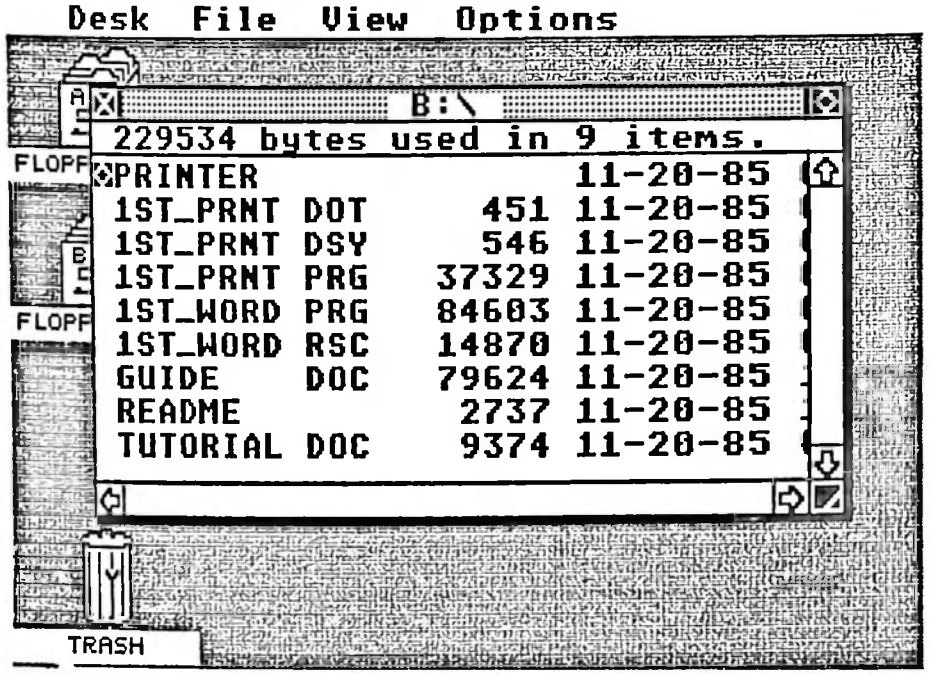


Figure 3.10-A list of the files on Drive B shown as text.

If you have used the Apple Macintosh, you know that the Mac's trash can is like a real wastebasket. If you throw something in it and then decide you want it back, there are ways to retrieve files from the trash can as long as the trash has not been "dumped." That is not the case with the ST's trash can. When you dump something in the trash, a dialog box asks you to confirm that you really want to get

rid of it. If you click OK in that dialog box, the file or folders you threw away are gone. Some people have suggested you think of the ST's trash can as an incinerator. If something is put into it, it is destroyed immediately. Using the ST's trash can calls for a little extra caution.

The finality of the ST's trash can is one reason why you should use the write-protect tab on disks that contain files you don't want to erase accidentally. If you try to delete files or folders from a disk that is write-protected, an Alert box will be displayed on the screen that says you cannot modify the disk because it is write-protected. If you want to delete files anyway, you must eject the disk from the drive, slide the write-protect tab down, and reinsert it in the drive. Then click Retry in the Alert box. If you don't want to delete files from that disk, just click Cancel. Alert boxes warn you about problems. Most of them have an OK button that must be clicked before you can continue working.

We have now discussed two of the four labels on the menu bar. Before going on, we should make two points again. First, in most cases all of the options in the drop-down menus are not active at the same time. Those in light type cannot be activated. For example, an option that operates on files in an open window will not work unless there is an open window with files in it.

In addition, the menu options we have discussed here will not necessarily be exactly the same as the options you have available in every situation. Different applications programs may add some options and omit others from the menus. It is also possible that Atari will modify the software that runs the ST and make changes in the labels on the menu bar or in the menus themselves. Therefore, don't be alarmed if your menus don't look exactly like the ones we've described here.

4

Exploring the GEM Desktop

This is the second of two chapters that deal with the basic principles of operating the Atari ST. If you are a novice at personal computing, these chapters give you a feeling for the way the ST relates to you. The knowledge you acquire is a solid foundation upon which to build a pleasant, positive, lasting partnership between yourself and the ST.

To round out your knowledge, this chapter deals with four topics -- the DESK menu options, windows, the mouse, and the disk drive. Some of the information has not been presented before, some has been discussed briefly in previous chapters, and some may be old hat to you by now. With that mix of content, we hope it will serve as a tutorial for new concepts. We also propose it as a reference that you can refer to when you need to check on details of operation.

All the topics discussed in this chapter are aspects of GEM, the graphics shell used by the ST to make it easier to deal with the TOS operating system. The way GEM is used on the ST is somewhat unique, but GEM is also available for several different computers, including the IBM PC. Although there are many differences between the IBM PC and the Atari ST, if you use GEM with both, the systems will appear to be very similar in the way they accept and execute commands for the operating system. Digital Research even has several applications programs that run under GEM and use the same type of graphics or icon-oriented environment. GEMwrite, a word processing program, and GEMdraw, a program that helps you create graphics, should be available in versions for both the IBM PC and the Atari ST by the time you read this. With programs like GEMwrite that are available for the ST as well as other popular computers, you could use an IBM PC at work and an Atari ST at home and still use the same operating system and applications software much of the time.

This chapter deals with the two remaining labels on the Menu bar, Desk and File, and it presents additional details on the GEM desktop with an emphasis on the way windows work.

DESK

A nonelectronic desk has accessories such as pens and notepads, and you can rearrange items on the desk or make personal adjustments such as lowering or raising the height of your chair. The GEM desktop also has accessories, and there are ways to make adjustments as well. To begin exploring the DESK label

on the menu bar, boot the computer with the Atari TOS System Disk in Drive A. When the window for that disk opens on the screen, click on the close box to remove it from the display. Now you should have a desktop that looks like the one in Figure 4.1.

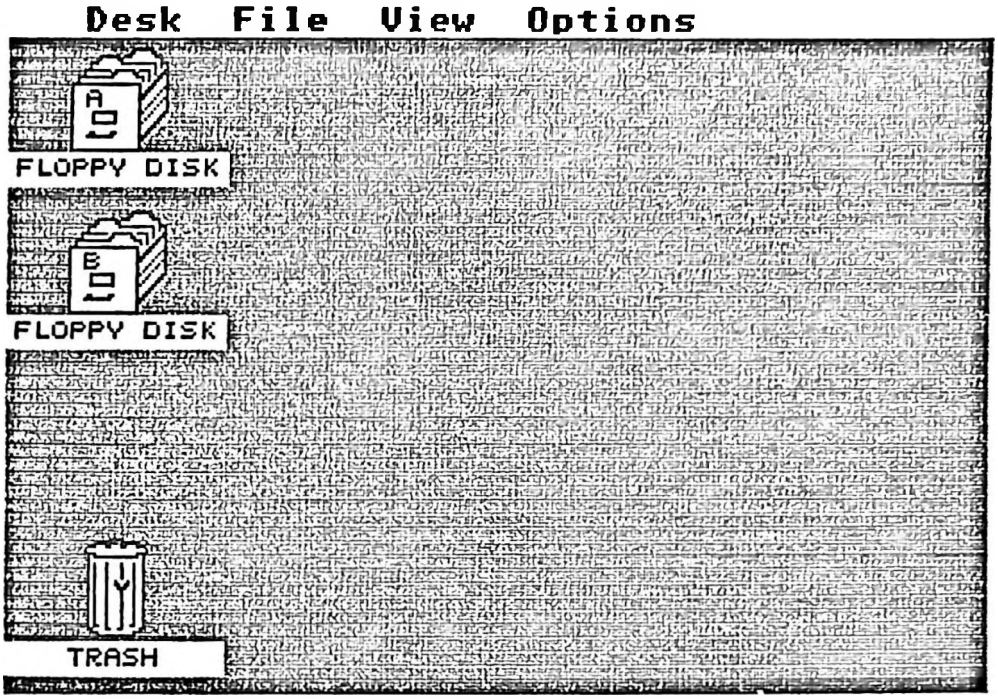


Figure 4.1-The GEM desktop with no windows open.

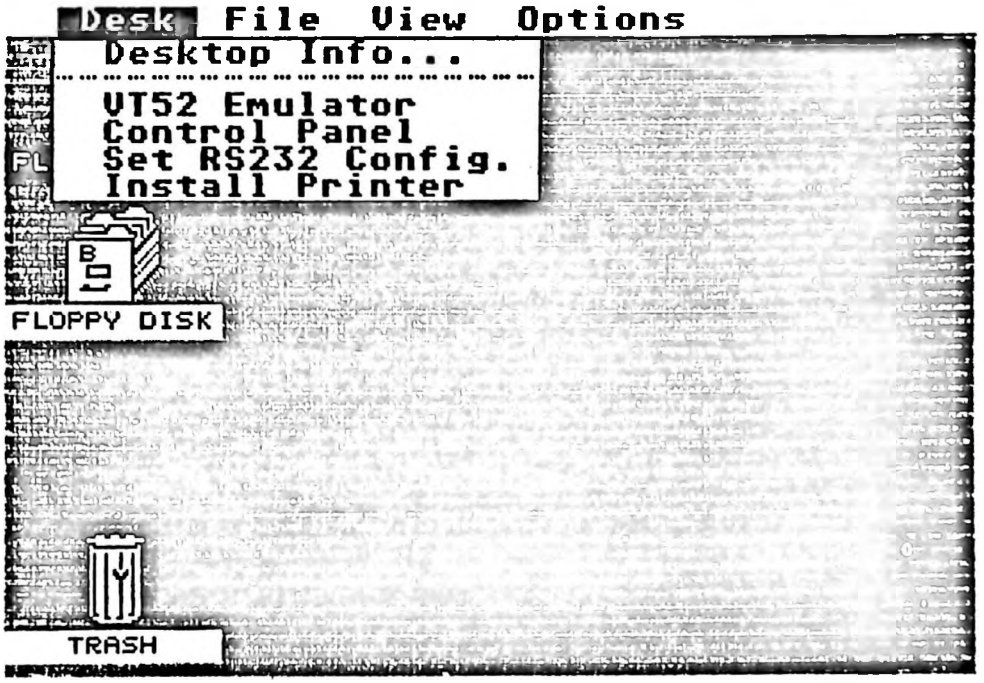


Figure 4.2-The DESK menu.

Now point to the DESK label on the menu bar. A menu will drop down that lists several items on it. The DESK menu is shown in Figure 4.2. We will explain each of the items on that menu in this section.

Desktop Info

If you select this option by moving the cursor over Desktop Info and clicking the left button, a dialog box will appear on the screen that tells you this is a GEM desktop and that it is copyrighted by Atari and Digital Research in 1985. You must click on the OK to remove this dialog box from the screen. The Desktop Info option does nothing other than display the dialog box. As new versions of GEM and TOS are developed, the information in the Desktop Info box may become very important because some ST software might work only with specific versions of GEM or TOS.

Control Panel

In contrast to Desktop Info, the Control Panel option on the DESK menu has many options. They let you customize the way many aspects of the GEM desktop operate. When you select the Control Panel option of the Desktop Info menu, the screen displays the control panel window which is illustrated in Figure 4.3.

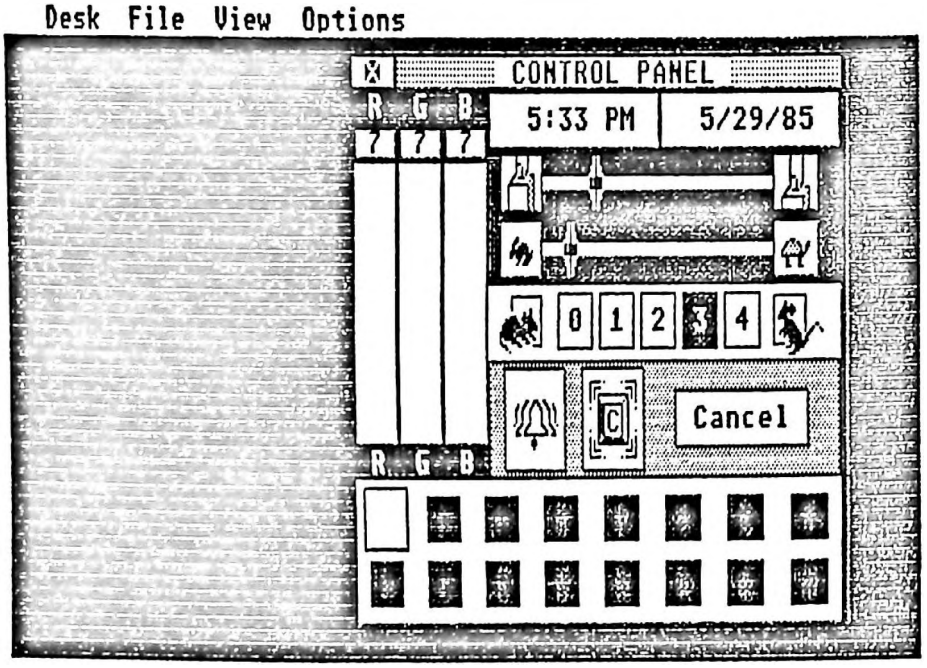


Figure 4.3-The Control Panel.

Most computers are take-it-or-leave-it machines. If you don't like the way some aspect of the machine operates, that's your problem. Not so with the ST. Select the Control Panel option from the DESK menu and you are presented with a complex-looking window of symbols and numbers. Each of those symbols controls some aspect of ST operation, and you can adjust the mode of operation to suit your personal preferences.

There are five different sections on the Control Panel: the clock/calendar, the keyboard response selector, the mouse click response options, audio feedback control, and palette/color selector control. Each of them is discussed below.

Time and Date. The date and time are shown in a rectangle at the top of the control panel. You can change the time and date by pointing to the time or date and clicking the left mouse button. When you click on the time or date, the window with that information changes to reverse video and you can press the backspace key to remove the current time or date. Then you can type the correct data. For example, if the date were listed as 5/29/85 and it is really July 15, 1987, you could select the date by clicking on it and then pressing the backspace key to remove the default date. Note that each time you press the backspace key, one digit in the date is erased. Then you would type 7 15 87--the slashes need not be typed. When the date is correct, click on the new date and it will change from reverse video to regular type and your new date is set. Use the same procedure to change the time. If the time is shown as 5:37 PM and it is really 1:15 AM, you would select the time by clicking on it and then use the backspace key to remove the current time. Then you would type 1 15 A. The ST fills in standard parts of the time such as the : between 1 and 15 as well as the M in AM or PM.

Keyboard Response: Key Repeat Control. Each of the keys on the ST keyboard is a repeating key. That means you can hold down the key and produce a series of characters from that key. The amount of time you must hold the key down before it starts repeating can be adjusted to suit your personal preferences. Just below the time and date is the electronic equivalent of a sliding switch volume control. To adjust the delay time before a key begins to repeat, place the pointer over the knob on that sliding switch and press the left button. As long as you hold the button down, moving the mouse right or left also moves the slide. When you release the left button, the slide control will move to the position of the pointer. The further left that slide is moved, the quicker a key begins repeating. Slide it to the right to increase the delay before keys begin to repeat. Some people prefer the repeat function to begin quickly, while others like a pause before a key pressed begins to repeat, and this option gives you a chance to set the ST keyboard up just the way you want it.

Keyboard Response: Repeat Speed. There is another slide control just below the one that determines how quickly a key begins to repeat. The second slide control determines how fast a key repeats. Move this control to the left to make keys repeat slower and move it right to make them repeat faster.

Mouse Click Response. Double-clicking the mouse is something most ST users do often because that is the way you open windows and issue some commands. You can control how quickly you must click the mouse to issue a double-click command by changing the numbers displayed just below the slide controls for the keyboard response selector. The number you click on determines how fast you have to double-click to get a response. Select the leftmost square if you want to click rather slowly and still register a double click. That square is adjacent to the resting mouse icon and has the number 0 in it. Select the rightmost square, which is adjacent to the standing mouse and has a 4 in it, if you want to register a double click only by clicking twice very quickly. Experienced ST users are often frustrated by settings of 0 or 1 because they may inadvertently issue a double click. They often select 3 or 4. The default setting is 3. To change the setting, point to the number you want to use and click.

Audio Feedback. Just below the double click control are three icons, a bell, a keytop labeled C, and a box with Cancel written in it. The bell and keytop control the type of audio feedback the ST provides.

Normally the ST makes a small clicking sound each time you press a key. That is reassuring to some people and annoying to others. If you do not want to hear a "click" each time you press a key, move the pointer over the keytop with the C on it and click the left button on the mouse. The keytop will change to a faint outline shape and the ST will be silent when you press a key. This option, like many others, is a toggle switch. Click the C-keytop once and it changes to a faint outline and clicks are off. Click the C-keytop again to toggle keyclicks back on and display the keytop normally. The volume of keyclicks, by the way, is controlled by the volume control on your video monitor.

The ST also provides another type of audio feedback. If you make an error such as striking a key the ST does not understand or pressing the wrong button on the mouse, the computer outputs a little burbling sound that signals an error. When the bell icon is displayed normally, errors generate audio feedback. Click on the bell, and it will change to a faint outline. That means you will not hear the error sound. Click on the bell again, and it toggles the error sound back on. As with the key clicks, the volume control on the monitor controls how loud the signal is.

Cancel Box. As you experiment with different settings on the control panel, there may be times when you decide you want to forget all the changes and reset all the controls to the settings they had before you began experimenting. To do that, move the pointer to the word Cancel and click the left button on the mouse. Everything will return to default settings except the date and time.

Palette Controls. The last part of the control panel to be discussed determines the colors used on the desktop. Palette controls are of interest only if you have a color monitor. They have no effect on a monochrome monitor.

In the low resolution graphics mode, the image displayed on the screen is 320 by 200 pixels. In that mode, the ST can display any of 16 different colors on the screen. Those 16 colors are displayed on the bottom of the control panel. They range from a solid white to solid black with several reds, blues, greens, and greys available as well. Think of these colors as 1 through 16. If you have a color monitor, the palette at the bottom of the control panel will have two lines of eight colors. Here is a list of those colors with numbers assigned to them:

1-White	5-Grey	9-Black	13-Lt Black
2-Red	6-Med Blue	10-Lt Green	14-Lt Green
3-Light Blue	7-Dark Pink	11-Dk Blue	15-Dk Purple
4-Rust	8-Army Green	12-Dk Green	16-Yellow

Unless you change them, these are the 16 colors the ST can use. Those are not all the colors the ST is capable of displaying, however. The ST's color palette has 512 different colors in it, but only 16 can be used in the low resolution mode. The 16 colors displayed at the bottom of the control panel are the default colors, but you can change them easily.

Note that there are three slide controls on the left side of the control panel. The three controls are labeled R, G, and B. That stands for Red, Green, and Blue which are the three primary colors. Color televisions, and color computer monitors, mix these three primary colors to produce all the other colors. Note that the box for color number 1 is larger than the other sixteen at the bottom of the control panel. At the moment, that is the selected color, and the settings on the three color slide controls tell you how that color is mixed. White is produced by setting all three slides to their maximum setting of seven.

Now move the pointer to the box with color number 6 in it, a dark blue. Click on that color and the blue box will enlarge to show it is now the selected

color. Note also that the slide controls for Red, Green, and Blue have changed. They reflect the settings that produce that color. Blue is set to 5, Green is 3, and Red is 0. Now move the pointer to color 11 in the bottom row. That is another blue. Click on color 11 and look at the slide controls for Red, Green, and Blue. Blue is again set to 5, but both Red and Green are set to 0.

Now move the pointer over the slide control for Blue and hold down the left button. Move the slide up to the 7 setting and release the left button. Color 11, the one that is currently selected, changes to a lighter blue because that color is now set to Blue-7, Red-0, and Green-0. Point to the Blue slide control again and hold down the left button. Move it to 0 and release the button. Color 11 changes to black which is what you get when all three primary colors are set to 0.

Move the pointer to the word Cancel on the control panel and click to reset the colors to their default values. Any of the 16 colors available in the low resolution mode can be changed by selecting the color and then adjusting the slide controls for Red, Green, and Blue. Then, when you program a color, it will be displayed according to the settings you provided unless the program resets them itself.

The GEM desktop uses some of the colors itself. For example, Color 1, which is normally white, is the background color. Unless something else is displayed, the screen will be white. Normally the border around the edges of the GEM desktop is white or Color 1. Color 10, a light green, is used for the main desktop area, and Color 9, a pure black is used for text and for the graphics that make up the control panel. Select Color 9, black, by pointing to it and clicking. Then move the Blue slide control up to six from the default setting of 0. When you release the button, most of the control panel, which is drawn in color 9, will be a pale shade of blue.

To experiment some more, move the Blue back to 0 and then select Color 10, the light green used for most of the desktop. Move the Red slide control up to 7 and when you release the button, the color is a dark yellow. Since color 1 (white), 9 (black), and 10 (light green) are the most often used colors on the GEM desktop, you could adjust those colors and the desktop would look very different.

Remember that any changes you make in the color palette can be cancelled by clicking on the cancel button. If you want to keep them in effect for the current work session, you can click on the close box of the control panel and the panel will be removed. Your settings will remain in effect until the computer is switched off. If you want to make your new settings the default settings for the operating system on the disk in Drive A, select the Save Desktop option on the Options menu. If you do that, the settings on the control panel will be used any time you start the computer with that disk.

Thus far, our discussion of the color palette has assumed you are working in the low resolution mode. If you work in the high resolution mode with a monochrome monitor, none of the color palette options will work because that monitor displays only one color. The default mode for the color monitor is low resolution with 16 colors, but it can also work in a medium resolution mode (640 by 200), but only four colors can be displayed. If you select the medium resolution mode through the Set Preferences menu of the Options label, the control panel will still display sixteen color boxes, but all of them will be black except 1, 2, and 10. The four colors the computer uses in the medium resolution mode are 1 (white), 2 (red), 9 (black), and 10 (light green). You can change any of the other colors, but they will remain black on the screen. Only four colors, 1, 2, 9, and 10 are used. With that limitation, setting colors for the medium resolution mode is the same as setting them for low resolution.

This concludes the tour of the control panel. If you want to keep the settings you have made, click on the close box. If you want to cancel them, click on Cancel first, then close the control panel window by clicking on the close box. If you want to permanently change the settings on the control panel, click on Save Desktop in the Options menu.

Install Printer

If you have a printer and want to use it with the ST, you may need to change some of the options in the Install Printer menu. You can display the menu by clicking on Install Printer on the DESK menu. A box like Figure 4.4 will be displayed. It lets you specify the type of printer you are using and provide necessary information such as which port the printer will be connected to. Each of the items on the menu is discussed below.

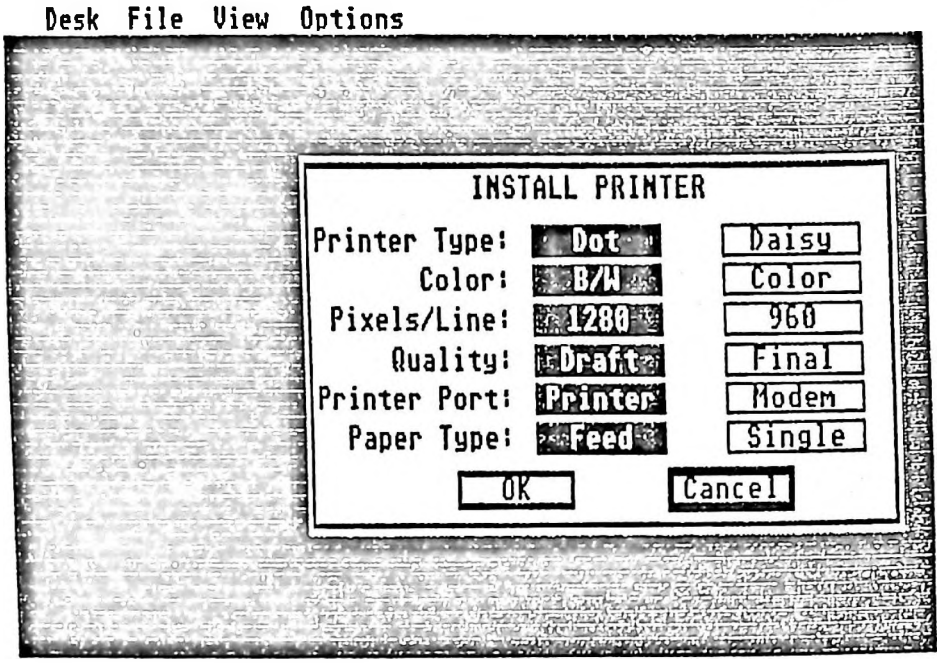


Figure 4.4-The Install Printer dialog box.

Printer Type: Dot or Daisy. The first item lets you indicate which type of printer you have. Most people use a dot matrix printer such as the Epson FX-80 or the Gemini SG-10 with the Atari ST. Note: Printer types are discussed in more detail in the chapter on I/O and accessories.

The default setting for Printer Type is dot matrix, so you need not make any changes here if you have a dot matrix printer. If you will be using a daisy wheel printer such as the Diablo 630 with the ST, point to the word Daisy and click on it. That will shift the reverse video from Dot to Daisy. Now the ST knows you are using a daisy wheel printer instead of a dot matrix model.

Color: B/W or Color. Most printers use only one color, but there are a few that can print in several different colors. An inexpensive model that works with the ST, for example, is the Okimate 20 from Okidata. If your printer is a standard one that prints only one color, leave this item set to B/W for black and white. If you have a color printer, click on the word Color.

Pixels/Line: 1280 or 960. If you have an Epson printer such as the FX-80, or a printer that works much like the Epson dot matrix model, set this option to 960. That works fine with Epson models that use 8-1/2" wide paper and print 10 characters to the inch. In fact, the 960 setting works correctly with all the printers we tried, but there are some high resolution printers that can print 1280 dots per line on paper that is 8-1/2 inches wide. If you have a high quality dot matrix printer such as the Toshiba 1041, you might try the 1280 pixels per line setting. On most printers, selecting this option causes the printer to cut off the right side of the screen image.

Quality: Draft or Final. If you select Draft mode, the printer will complete its work much quicker, but the quality of the printout will not be that good. When you want the highest quality you can get and have the time to wait while the printer works, click on Final. That produces a higher quality of print but it takes longer.

Printer Port: Printer or Modem. Printers with a standard Centronics-style parallel interface can be connected to the ST through the printer port on the back of the keyboard. Most printers use a parallel interface, and that is the default setting since the Printer port on the ST is really a parallel port.

If you have a serial printer, you will have to connect it to the ST through the Modem port. That generally calls for a serial cable and a device called a null modem since the Modem port on the ST is a serial RS-232 port, but it is wired for use with a modem. A regular serial cable can be used to connect the printer to the modem port, but you will need a null modem device either built into the cable or added to one end because some of the wires must be switched in the cable. Printers and modems are not wired the same even though they use the same type of cable and connectors.

Paper Type: Feed or Single. Some printers can only deal with one sheet of paper at a time. Others have tractor feeds, sheet feeders, or some other mechanism that allows them to print one sheet after another. If your printer can use continuous feed paper or has a sheet feeder, the default setting of Feed will be fine. If your printer can handle only one sheet at a time, select the Single option, and the ST will stop after each page is printed to let you insert another sheet in the printer.

OK or Cancel. When you set the options in the Install Printer menu, click on OK to tell the computer you have finished. The menu will disappear. Select Save Desktop from the Options menu to make these settings the defaults for the GEM desktop. If you decide to forget about making any changes in the Install Printer menu, click on Cancel and the menu will disappear.

Set RS232 Configuration

If you have a serial printer, you will probably need to set some items on the Set RS232 Configuration menu. Like the Install Printer menu, it is accessed from the DESK drop-down menu. The serial port on the ST uses the RS232 standard, which is what most serial ports on computers and printers use. This menu lets you set things like the baud rate and parity, and the settings for the ST must match those for the printer before you can use a printer with a serial port.

The Modem port on the ST is actually an RS232 serial port you can use for connecting a printer or a modem. There is more information on this port as well as directions for setting the items in the Set RS232 Configuration menu in the telecommunications chapter.

VT52 Emulator

That Desk label also has a menu named VT52 Emulator. It can be used to set the ST up as a terminal for accessing mainframe computers. The VT52 termi-

nal is a popular terminal made by Digital Equipment Corporation, and the items in this menu let you set the Atari ST up so a large computer thinks it is just another VT52 terminal. This menu is also discussed in more detail in the chapter on telecommunications.

WINDOW ANATOMY

A fundamental element in the GEM desktop environment is the window. You have worked with windows all along, but we have not yet discussed some of the finer points of window operation. Windows are a simple concept, but the execution of that concept is somewhat complex. Just as each word in a sentence adds depth and richness to the meaning of that sentence, each element in a window adds information or a means of controlling what the computer does.

Perhaps the best way to explain the anatomy of a window is to fire up the ST and play with a window. Find the disk for 1st Word, insert it in disk drive A, and switch the system on if it is not already running. (If you do not have a disk for 1st Word, any other disk with several files on it can be used to illustrate the points in this section.) Push the reset button on the back of the ST beside the power connector, if it is already on. When you do that, the computer will load a few files that tell it how to set up the desktop, and it will automatically open a window from the disk in drive A. The screen will have the icons for disk drives A and B, the disk A window, and the ever-present trash can in the bottom left corner.

If the window for this disk doesn't open automatically, move your pointer over the Floppy Disk A icon and double-click to open the window for that disk. Now let's take a close look at the window.

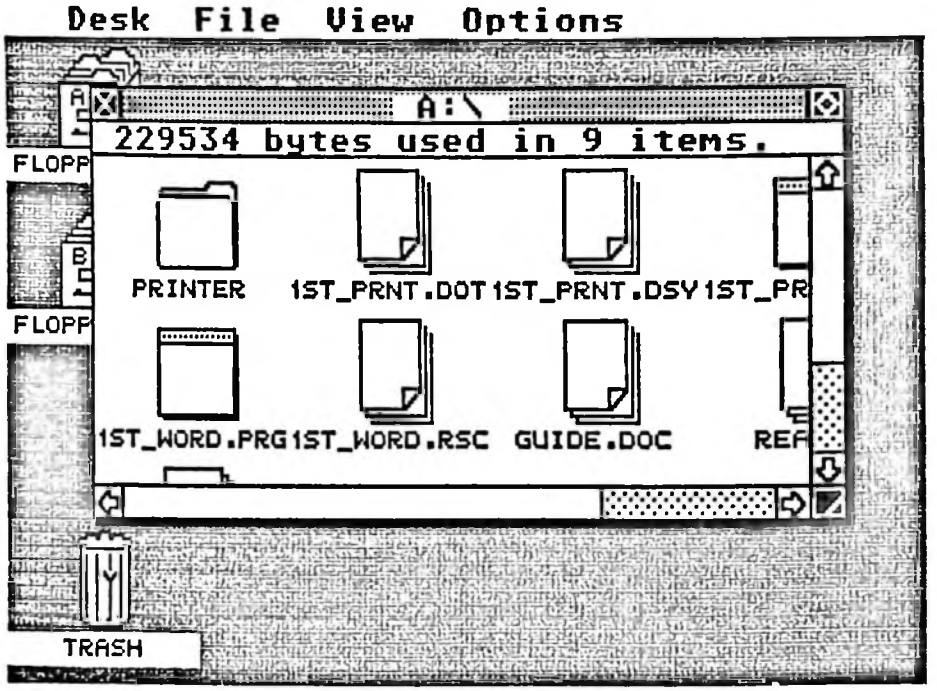


Figure 4.5-The window for the 1st Word disk.

Move Bar, Close Box, Full Box, and Disk Identifier

Several closely-spaced lines run across the top of the window. These are the Move Bar. On each end of the title bar are two other window elements. The box on the far left is the Close Box. You can close the window by clicking once in that box. On the right end of the Move Bar, there is another box, the Full box. If you want the active window to fill the entire screen, you can click on the Full Box. Click on the Full Box again and the window will return to the size it was before you clicked. You might experiment with the Full Box now to get an idea of how it works.

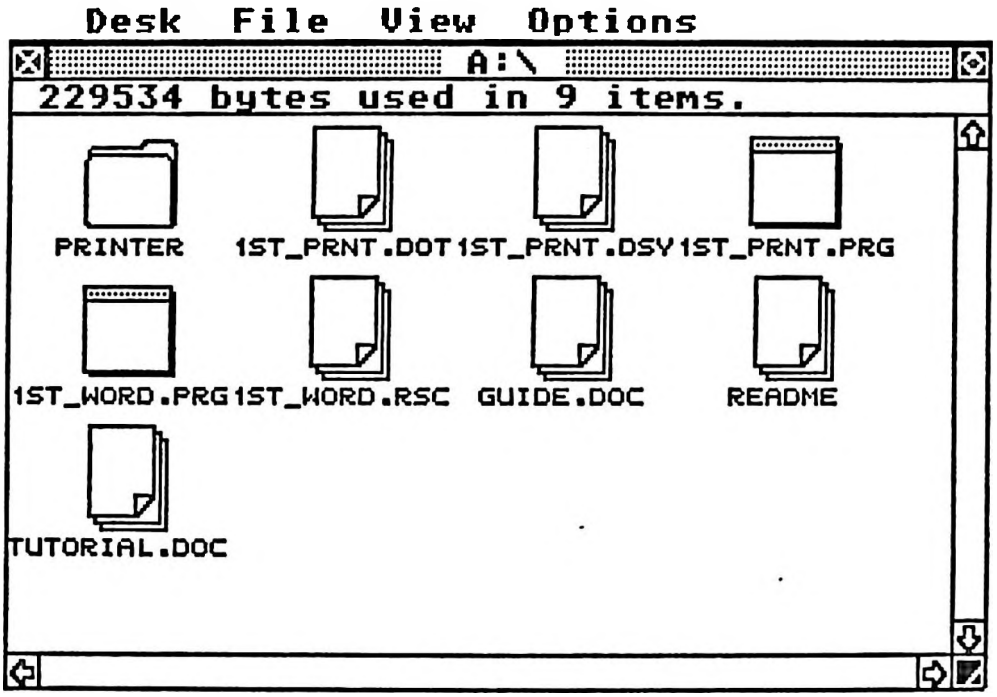


Figure 4.6-Clicking on the Full Box expands the window to fill the entire screen.

The other element in the Move Bar is the Disk Identifier, which is located in the middle of the bar. The Disk Identifier is currently A:\. If the window were for the disk in Drive B, the identifier might be B:\. If the active window were for a folder of files named PRINTER from a disk on Drive A, the disk identifier would be A:\PRINTER\. The Disk Identifier thus tells you where the window of information came from.

Information Line

Just below the title bar in the window is a line that has several pieces of information. On the window for the PRINTER folder, the line told us there were seven items in this folder that take up 55,477 bytes of space on the disk. Remember that this information is for one folder of files. If there are other files or folders stored on this disk, there may be much more than 55K of space used. Close the window for the PRINTER folder and the window for the disk in Drive A will be visible again. The Information Line for this window tells us 229,534 bytes of space on the disk have been used for nine items. Since we have single sided drives in our ST system that can store 360K on a disk, there is around 130K of space available.

As you store data on a disk, you may want to keep an eye on how much space remains on a disk. It is easy to fill up a disk without realizing you are running out of space.

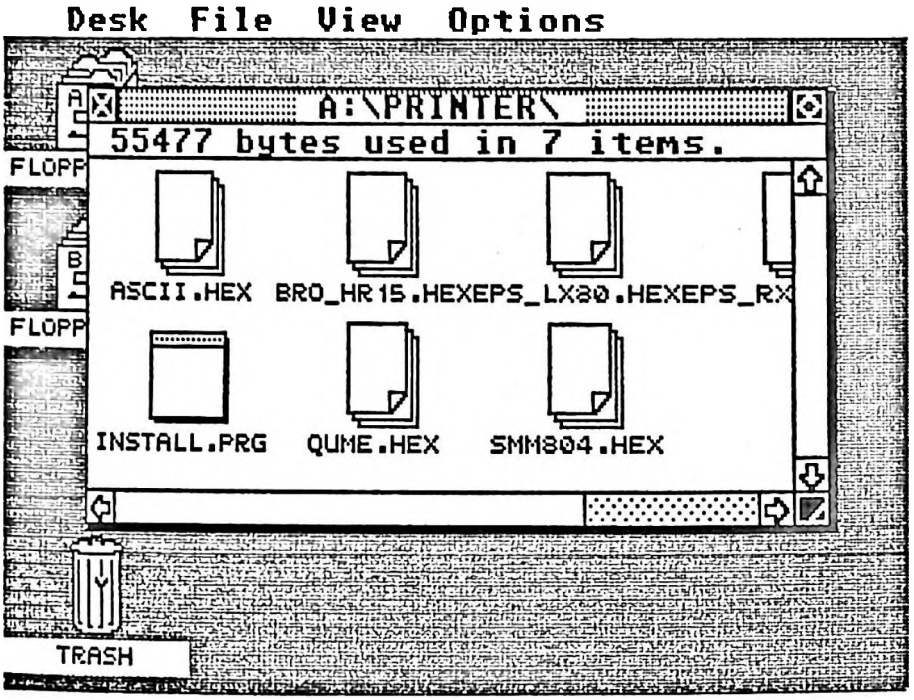


Figure 4.7-The window from the folder named PRINTER on Drive A.

Note that the line says there are nine items in this window. On other computers that might mean there are seven files on the disk. You cannot equate the term items with the term files when working with the ST, however, because the ST can group files into folders. A particular window might thus have eight files and one folder, but the folder may have a number of files in it. Nine items might actually represent many files, since some of the files could be in folders.

Making a Window Active

Another aspect of window operation is activating a window. If you want to open a window from a disk icon, move the pointer to that icon and double-click. The window for the disk in that drive will open and become the active window. You can tell a window is active by the Move Bar. It will be shaded with lines. Observe the two windows on the screen. One is from Drive A and one from Drive B. You can tell the window from B is active because it is displayed in front of the other window and because the Move Bar is shaded. Note that the Move Bar on the A window is clear.

With two windows displayed on the screen, you can make another window active by pointing to the Move Bar of the window you want to make active and clicking once. Do that on window A and it will become active as shown in Figure 4.8.

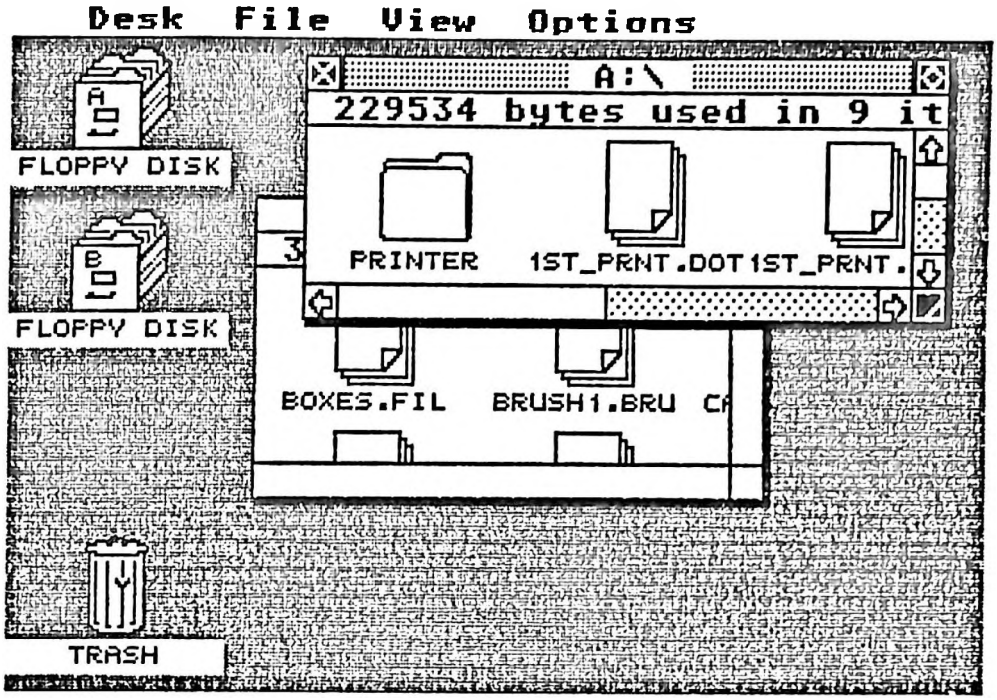


Figure 4.8-After clicking on the Move Bar for the A window, it becomes active.

Dealing With Files and Folders

Just below the information line is the main body of the window. Icons for each of the items on the disk are displayed in that window. There are several things you can do with the icons in the window.

Rename. You can change the name of a file or folder by moving the pointer over the name. The pointer changes to the I-beam that indicates you are working with text. You can then change the name the same way you changed the name of the disk. Remember, however, that you cannot rename items that are locked. Check an item's status by selecting Get Info in the file menu after making the item active.

Select. You can select an item by clicking the pointer over it.

Open. You can open an item in the active window by double-clicking the pointer over it or by selecting Open in the File menu. Opening a folder would cause another window to appear on the screen to display the contents of that folder. Opening an applications program tells the computer you want to begin using that program, and opening a document created by an applications program tells the computer you want to work on that document.

Dragging Icons

Move. Items in the window can also be moved about. This procedure is called dragging. To illustrate dragging, close the window for disk A and remove the 1st Word disk from the drive. Then place a copy of the Atari TOS System disk in Drive A and open a window that displays the contents of that disk. Click Full Box on the window to enlarge the window.

Suppose you have a file named DESKTOP.INF whose icon is displayed in the top left of the window, but you want the icon for that file to be located in the bottom right corner of the window. Move the pointer over the icon and hold down

the left button on the mouse. Then move the pointer to the lower right of the window and release the switch. As the pointer moves, an outline of the DESKTOP.INF icon moves too. That is dragging. When you release the switch, the icon moves to the place where you released the switch. You can move icons around in the window, you can move them to the trash can if you want to throw them away, and you can move them into or out of folders.

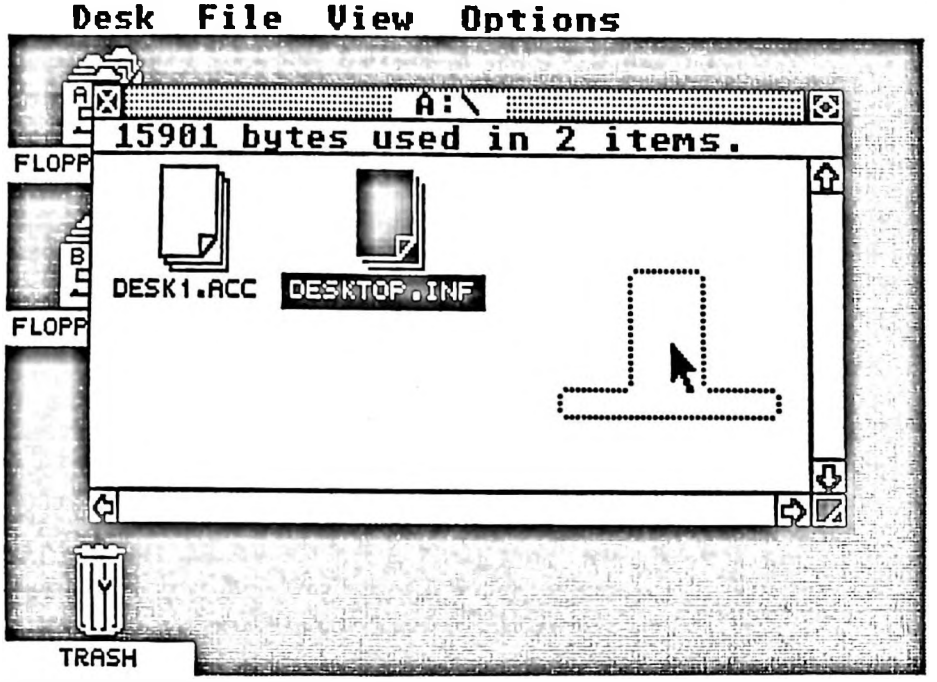


Figure 4.9-The DESKTOP.INF icon is being moved.

Creating Folders and Moving Files

To move two files into the empty folder, for example, open the folder so its window is on the screen. Then drag the two files into the folder's window. That places the files in the folder, much as you place documents in file folders in a filing cabinet. One problem with moving files into folders is that the folder window may obscure the window where the files are located. Fortunately, there is a way of dealing with that problem.

Moving the Window. The window for disk A is probably at the top of your screen. If that is not where you want it, the window can be moved. Place the pointer anywhere along the Move Bar at the top of the window, press and hold down the left button on the mouse, and then drag the pointer. As the pointer moves, you see an outline of the window moving too. Wherever you release the switch is where the window will be moved to. You can even move the window so only a little of it shows on the screen. Moving windows will be very handy when you have three or four on the screen at once.

When you move a window and then close it, the next time you open the window, it will go to the last location instead of the default location. In fact, if you move a window, close it, and open another window, the second window will also move to the last location of the window you closed.

Changing Window Size. Sometimes, it isn't enough simply to move a window. It may still be too big or too small to work with. We have already discussed the Full Box on the right end of the Move Bar, but there is another way to change window

sizes, the size box. In the bottom right corner of the window is a dark box with a diagonal line across it. Put the pointer in the size box, press and hold down the switch on the mouse, and you can reduce or increase the size of the window by moving the pointer. Move the pointer up and the window's height decreases, move it to the left and the width decreases. You can enlarge the box by moving the pointer to the right or down.

Being able to change the size of a window and to move it gives you a great deal of flexibility. However, when you reduce the size of a window, there may not be enough space for you to see all the items in the window. Again, there are ways of dealing with that problem.

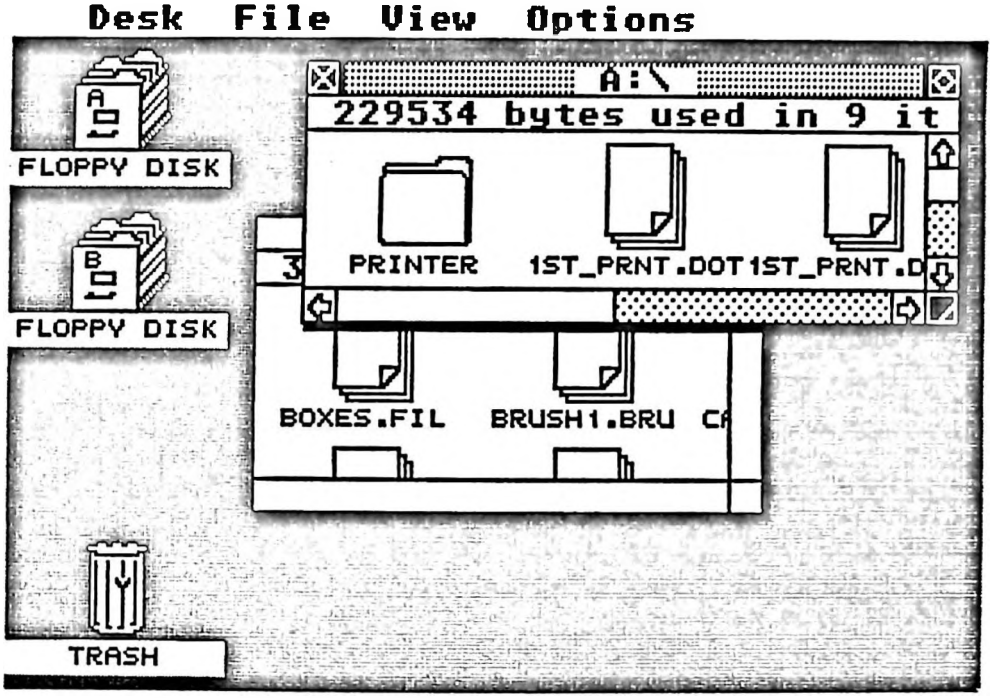


Figure 4.10-The window for A has been reduced using the Size Box.

Scrolling the Window. If you've been playing with the window, use the size box to make it approximately the size it was when it was opened. Now find the four arrows on the boundaries of the window. Two are near the size box (down arrow and right arrow) in the bottom right-hand corner, one is on the top right (up arrow), and one is on the bottom left (left arrow). Now make the window so small that only a few of the icons in the window are visible. Reduce both the height and width of the window. Now look carefully at the borders of the window. Much of the area between the arrows in the border changed to a shaded pattern, and there is a box in the checkered area. The checkered area is called the scroll bar. If you place the pointer in the scroll box (the box on the scroll bar) and move the pointer along one of the scroll bars, the area of the window you can see will change as you move the scroll bar.

A faster way to scroll is simply to place the pointer somewhere on the scroll bar and click once. The part of the window you can see will change automatically. In addition, you can click the pointer in any of the four arrows in the border of the window and scroll the window. Try working with these three ways of scrolling a window for a while to get used to the way they work.

In the window we are using, both scroll bars, on the bottom and the right, should be checkered and operational. However, if the height of a window is large enough that it does not interfere with the display of items in the window, the scroll bar on the right side will not be operational. The same is true of the scroll bar on the bottom of the window if the width does not hide items. The scroll bars are only active if they are needed for you to view all the items in the window.

DEALING WITH DISKS

Throughout the last two chapters, you've been using the disk drive in the ST, but we haven't really discussed disk operation in any detail. In this section of the chapter, you will learn about how the disk drives work. You'll be shown how to prepare a disk for use in the ST. You'll also learn how to put copies of files and folders on a disk as well as how to make copies of an entire disk.

Some Disk Basics

The ST uses a specially designed 3-1/2" disk drive. Inside the protective plastic housing of each disk is a thin platter of plastic that spins when the disk is inserted in the disk drive. The first drives available for the ST, the SF354, could record to and read from only one side of the disk, but the SF314, a double-sided drive, is now available. The examples in this chapter were done on a single-sided drive, but the major difference between the two types of disk drives is the amount of data they can store on a disk.

The recording head in the disk drive takes data sent to it by the computer and places a magnetic pattern on the disk that represents that data. Since the read head must be able to find the magnetic patterns on the disk and convert them back into data the computer can interpret, the disk drive is a very precise mechanism. It divides the surface of the disk into 80 concentric circles. (Double-sided drives divide each side of the disk into 80 circles.) Each circle on the disk is called a track. Thus, if the computer tells the disk drive to look at data in track 40 on the disk, the disk moves the read head to the middle of the plastic platter because that is where track 40 is located.

Each track on an ST disk is divided into sectors. There are nine sectors in each track, and a sector can store 512 bytes of data. On the ST computer, you can store 4608 bytes of data in each track (512 times 9), and a single-sided disk can hold up to 368,640 bytes of data. This is often referred to as 360K since 360 times 1024 (1K) is 368,640. A double-sided drive can store 720K of data.

Formatting a Disk

Note: All the examples in this section assume you are working with a one-disk-drive ST. If you have two drives, the instructions on swapping disks can generally be ignored, since the ST will automatically move from one disk to another as needed.

Although Atari sells 3-1/2" disks for the ST, you can use disks from many different manufacturers. Some cheaply made, off-brand disks could cause you trouble (e.g., by leaving a film of oxide on the read/write heads of the disk drive), but most name-brand disks that are ST-compatible will work fine.

A raw disk right out of the box is not ready to work, however. The computer needs to draw a magnetic map of the tracks and sectors on it before you can store programs or data files. This process is called formatting the disk. If you put a new disk in the drive, and try to open the window for it, the drive will try to

read it, and then an alert box will tell you there is a problem. The alert box may say there is a problem with the drive itself, but the fault is usually in the disk. The drive does not know what to do with a disk that has not been formatted for the ST. Click on the OK to remove the alert box from the screen.

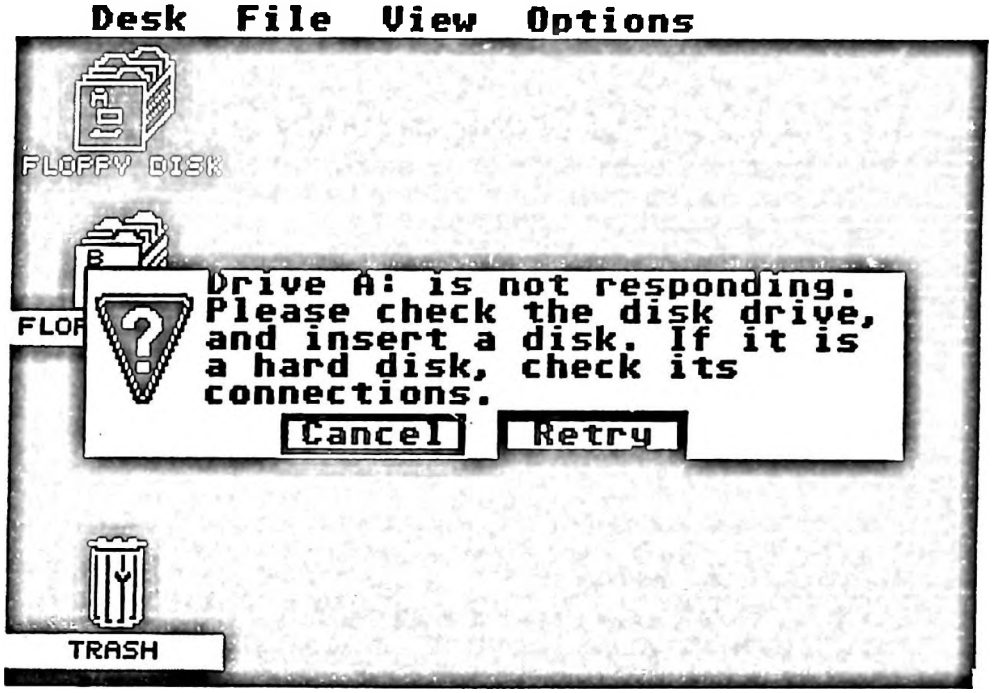


Figure 4.11-This Alert Box may indicate the disk you are trying to use has not been formatted.

To prepare a new disk for use, you insert it in a disk drive, and then point to the File label on the Menu Bar. The File menu has several options on it, but the one you want to use now is the last one, Format. Select either the Floppy Disk A or the Floppy Disk B icon. Then move the pointer to the Format option of the File menu and click once. That will begin the formatting process. An Alert Box is displayed on the screen that tells you formatting will erase all the data on the disk in drive A or B (whichever you selected before beginning the formatting process).

If a disk with data or programs is in the drive where formatting will take place, eject it by pressing the eject button on the front of the drive. When the disk is ejected, insert the new disk in the disk drive (or a used disk you want to erase and format again). Then click OK in the Alert Box. The computer will display another dialog box. This box has several options. The Drive ID will generally be A or B and indicates which drive will be used for formatting. If you want to label or name the disk, just type any name you like (e.g. DATA1). Whatever you type will be displayed beside the Disk Label heading.

Select your names carefully; try to select names that indicate something about the files on the disk (e.g., Letters1, Taxes85). The name can be as long as 8 characters, but long disk names can clutter up the screen. Select meaningful names, but don't go overboard (e.g., My Great American Computer Book2). If you make a typing error, use the backspace to erase the mistake. It is a good idea to give each disk a unique name since the computer sometimes stops operations and

asks you to insert a certain disk. If it identifies the disk by name and you have several disks with the same name, things can get confusing.

The double-sided disk drives (SF314) can format a disk on both sides or only on one. The SF354 drive can only format disks on one side. Click on the format option you want to use, single-sided or double-sided. The only time you might want to format a single sided disk on an SF354 drive is when you want to create a disk that can be used by someone who has only single-sided drives in their ST system.

With the preliminaries out of the way, click on FORMAT and the process will begin. Click on EXIT if you decide not to format a disk.

When you click on FORMAT, the ST will display a dialog box that tells you it is formatting the disk. As formatting progresses a long rectangle will gradually fill with shading. When it is completely shaded, formatting is complete. Another dialog box will appear that tells you how many bytes are free on the disk. Click on OK and the Dialog Box for Format will be displayed again. If you wish, you can remove the newly-formatted disk, insert another, and begin the formatting process again.

It is generally a good idea to have several formatted disks available for use. Why not get a box of 10 disks and format them for use? Only new disks need to be formatted, however. If you buy a program on disk, that disk already been formatted. In fact, if you format it, you will erase any programs or data stored on it.

We will be using the newly formatted disks to show you how to save files and folders so keep a few of them handy while you read the remaining sections of this chapter. One thing you should definitely do, however, regardless of the number of disks you format, is to eject each one after it is formatted and attach one of the stick-on labels that come with a box of disks. Then write the name of the disk on the label. The best time to write the name on the disk is right after you've initialized it. That helps to avoid confusion later.

What if the computer is turned off and you want to initialize a disk? If you simply insert a new disk and turn the ST on, it displays an alert box that says something is wrong with the drive. The ST wants nothing to do with raw disks right after you turn it on. You must first insert a disk with the system files on it. When the ST "boots" from a disk with TOS on it, you can can eject that disk and insert a blank disk to be initialized.

Copying an Entire Disk

Although the disks used by the ST are in a protective hard plastic shell, there is always the possibility of damage. Dust, electrical fields from motors, static electricity in the room, excessive heat in your car, and a thousand other predators are out to get your disks. It is therefore very desirable to make backup copies of important disks.

For example, you probably received a systems disk, a language disk, and a disk with 1st Write on it with your computer. You should make at least one backup copy of each of those disks. Two or three backup copies would be even better. The copies will be the working disks you use, and the originals should be stored away somewhere safe. Use them only when one of your working disks are damaged and you need to create another working copy. We generally make at least two copies of each program disk and one backup of every disk with important data files on it.

This section explains how to make a copy of an entire disk. We will use the system disk in the example, but the procedure is basically the same regardless of the disk you are copying. If you have ever made copies on an Apple Macintosh computer with one disk drive, the thought of making backups may be distasteful since some versions of that computer required you to swap disks as many as twenty

times to make a backup. You can relax because the method of making backups on the ST is easy and straightforward.

If the computer is not on, insert your system disk in Drive A and switch it on. This is the first step regardless of whether you are going to make a copy of the system disk or some other disk.

Then close the system disk window if it is open. Do that by clicking the Close Box. Now you should have a desktop with the trash can icon and icons for Floppy Disk A and Floppy Disk B on it. Eject the system disk from Drive A and insert the disk you want to copy. If it is the system disk you want to copy, you have already accomplished this step.

Put a formatted blank disk in drive B. Then move the pointer over the Floppy Disk A icon and drag it over the icon for Floppy Disk B. Dragging is accomplished by pointing to an icon and then holding down the left button while you move the pointer. As long as you hold down the left button, you will drag the icon with the pointer. When you release the left button on the mouse, the disk copy process will begin.

If the icon for Floppy Disk A isn't right over the icon for Floppy Disk B, try the procedure again and be sure to release the left button when the pointer is over the Floppy Disk B icon.

A dialog box will appear that tells you copying disk A to disk B will erase anything stored on the disk in Drive B. Verify that your disks are in the correct drives and then click OK. Click Cancel if you want to stop the copy process.

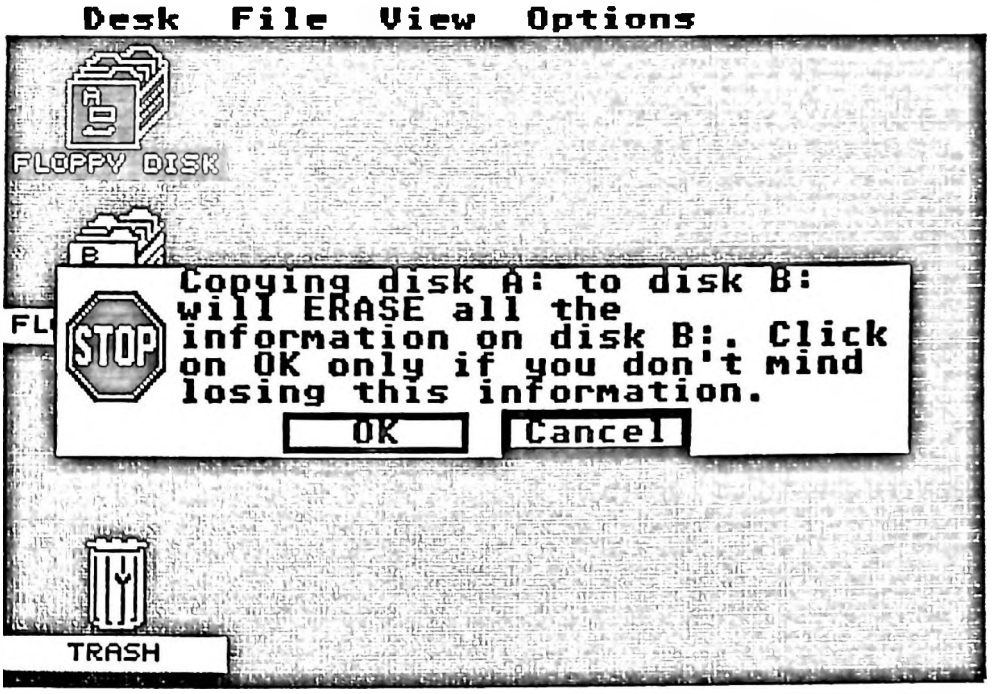


Figure 4.12-This Dialog Box will warn you that everything on disk B will be erased when you copy the contents of A to it.

When you click OK, the copy process will continue, and you need do no more than follow the directions displayed on the screen. Another dialog box will tell you the source disk (the one you want to copy) is A, and the destination disk is B. Click COPY if that is the way things should be. If you have two disk drives, the copy process will begin now and a dialog box will show you how much of the

source disk has been loaded into the computer and how much has been copied to the destination disk. When the copy process is complete, the Diskcopy dialog box will return to the screen.

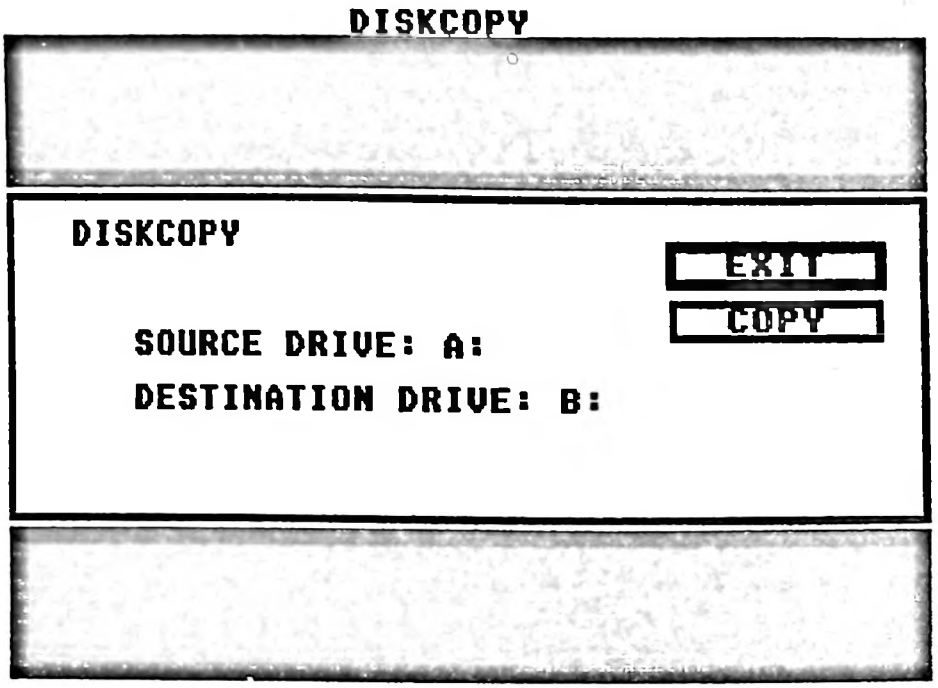


Figure 4.13-The Diskcopy Dialog Box.

If there are other disks you want to copy, you can insert another source disk in Drive A, a formatted destination disk in Drive B, and click on COPY to begin the process again. When you are finished, click on EXIT to return to the GEM desktop.

If you are making backups and have only one disk drive, the procedure is essentially the same, but the ST will display instructions for swapping disks. Carefully follow the directions such as "Insert Disk B into Drive A." Don't forget to click OK in the dialog box after you eject one disk and insert the other. You will have to eject the source disk or destination disk and insert the other a few times before copying is completed. Be sure to distinguish the source disk (the disk from which you're copying) from the destination disk (the disk to which you're copying). After several swaps, the copying process is completed, and you can click EXIT to return to the GEM desktop or make another backup if you wish.

We strongly suggest you make at least one more backup copy of the system disk and any disks with programs or important data on them as insurance against damage to other copies. Failure to make backup copies of important disks is one of the most common and most frustrating mistakes new computer owners make. When you're finished, click EXIT, and you will be returned to the GEM desktop.

Copying Files from One Disk to Another

There are times when you do not want to copy an entire disk. Instead, you may want to copy only one file or folder from one disk to another. That is easily accomplished using procedures you have already learned. Suppose, for example, that there is a file named LETTERS.DOC on the disk in Drive A, and you want to transfer that file to the disk in Drive B. To do that you would open the window for

disk A, find the LETTERS.DOC file, and drag it over the icon for Floppy Disk B. When you release the left button, the copy process would begin. The procedure is essentially the same as described above for copying all the files on a disk except you drag a file or folder icon instead of the icon for the entire disk. If you have one disk drive, the ST will tell you when to remove one disk and insert the other.

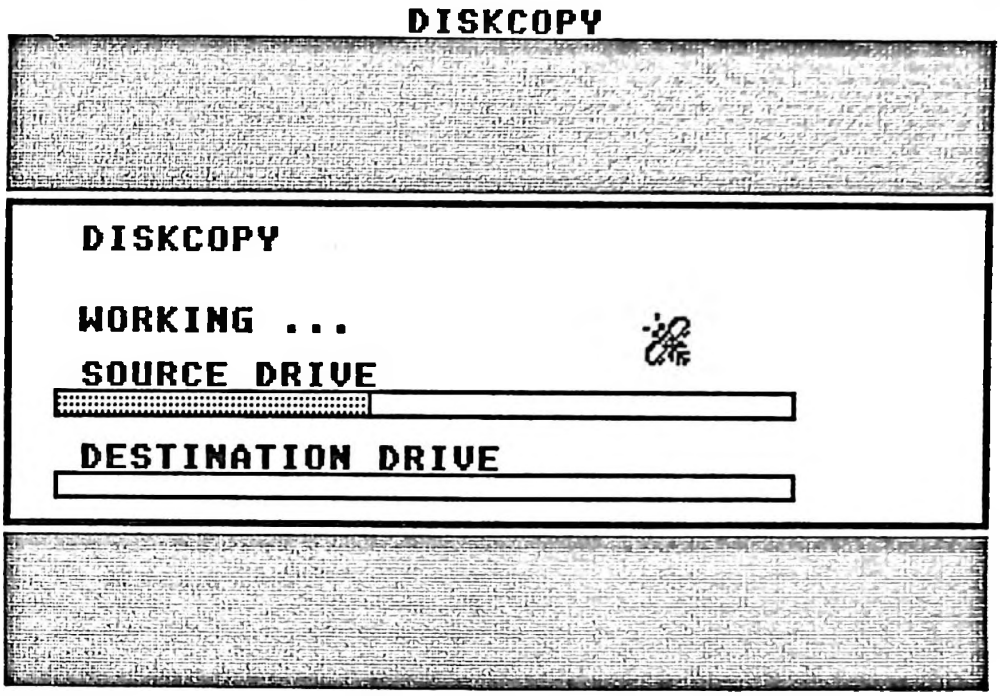


Figure 4.14-This Dialog Box indicates the progress of the diskcopy procedure.

Copying Files into a Folder

The copying procedures discussed thus far copy a file or folder to another disk. What if you want to place a file in a folder that already exists? Open a window that contains the destination folder, and open the window that contains the file you want to copy. Drag the file icon until it is over the icon for the folder and release the left button. The file will be copied into the folder. Follow the directions the ST displays on the screen. You can copy files into folders on the same disk or into folders on another disk.

This general procedure works for any type of file, so long as there is enough room on the destination disk to accept the new file. Note also that when you copy a file into a folder, you are not simply moving that file. That is, the current file is not moved from where it was to the new location in a folder. The file stays where it is and a duplicate of that file is created in the folder. That means you could have two exact copies of the same file on a disk, one displayed by itself and another in a folder.

Deleting Files

If you would like to remove a file or folder from a disk, open the window for that disk, find the item you want to delete, and drag it over the trash can. When you release the button, a dialog box will tell you how many files and folders will

be deleted and ask you to confirm the deletion by clicking OK. Do that if you want the ST to carry out the deletion, or click on CANCEL if you do not.

If you get a message that says you cannot drag folders, files, applications onto the desktop, that means you missed the trash can. It is a little like pitching paper into the trash can from across the room and having it land on the floor nearby. Click OK and then try again. Be sure to place the item to be deleted right over the trash can. Remember that the ST's trash can is really an incinerator; when you drag something into it, it is gone! You get no opportunity to rummage through the trash and find something you threw away accidentally.

Deleting and Copying Text Items

Items in a window are normally displayed as icons, but the Show as Text option of the View label will display files and folders as text. This was discussed in the previous chapter. You can also delete files and folders as well as copy files and folders when they are displayed as text. As with icons, you drag the item you want to copy to the location where the copy should be made. The process is the same as with icons.

Creating New Folders

The ST stores information in files that are stored on disks. The files themselves can be displayed on the screen as individual items with a name and an icon, or they can be organized into groups of files and placed in a folder. A folder is simply a convenient way of keeping track of several files that have a similar purpose. For example, you might keep all the data files that contain tax information in a folder named TAXES.

There is no folder named TAXES on the disks that came with the ST, but it is easy to create a new folder. Suppose the disk in Drive B contains six or seven individual files that contain tax information, and you want to create a folder where all those files will be stored.

Begin the process by opening the window for Drive B and making it the active window on the desktop. Then select the New Folder item in the drop down menu for File. A dialog box will appear on the screen where you can type the name of the new folder. Type the name you want to use for the folder (e.g., TAX DATA). Use the backspace key to correct any errors, and click on the OK button. Click on Cancel if you change your mind about opening a folder. When you click OK, a new folder will be created using the name you selected. Once it is created, you can use the procedures described above to move files into it.

THE FINER POINTS OF DESKTOP OPERATION

This chapter and the previous one have dealt with many aspects of working with the GEM desktop. As different aspects of GEM were discussed, we have occasionally omitted some details, or an alternative way of doing something, in an effort to make the explanations easier to follow. In this section we will discuss some of the points we omitted earlier.

Controlling the Mouse Pointer from the Keyboard

The standard way of moving the pointer around on the screen is the mouse. Roll the mouse around on your desk and the pointer on the screen moves too.

The "standard" way of doing things is not always the best way for everyone. If you are a fast touch typist, removing your hand from the home keys on the keyboard to use the mouse may actually slow things down. Fortunately the ST has another way to move the mouse. If you hold down the Alternate key and strike any of the cursor control keys, the screen pointer will move eight pixels in the direction of the cursor key you press. Other key combinations that involve the Alternate key let you perform tasks normally done with the mouse. Here is a summary:
Alternate-cursor key. This combination moves the screen pointer 8 pixels in the direction indicated on the cursor key.
Alternate-Shift-cursor key. This three key combination moves the screen pointer 1 pixel in the direction indicated on the cursor key.
Alternate-Insert. This key combination selects the icon the screen pointer is pointing to. It is does the same thing as pointing to an icon and clicking the mouse.
Alternate-Clr/Home. This key combination, which involves holding down the Alternate key and striking the key labeled Clr/Home, is the same as clicking the right button on the mouse.
Alternate-Insert-cursor key. You may have already guessed what this three key combination does. It lets you "drag" an icon from one location to another. The icon moves in direction of the cursor key you select. This is the same as pointing to an icon and then holding down the left button on the mouse while you move the mouse.

Selecting More Than One Item

In most situations, you will select one item or icon at a time by pointing to it and clicking. You can also use the key combinations discussed in the previous section. There are times, however, when you will want to select more than one icon.

That is a problem when you use the standard mouse method since you select an item by clicking on it. Then if you point to another item and click on it, the second item is selected, but the first item will automatically be deselected.

There are two ways to select more than one icon. The first, and often the easiest, is to place the pointer near, but not touching, the items you want to select and then press and hold down the left button. Don't release the button. Now drag the pointer across the icons you want to select. As you move the pointer, a rectangle of dashes will appear on the screen. Move the pointer so the rectangle encloses all the items you want to select. When you release the left mouse button, all the items inside the rectangle will be selected.

The point-and-drag approach with the left button depressed works well when the items you want to select are adjacent to each other, but it will not work if you want to select an icon in the top left corner of a crowded screen and another on the opposite side near the bottom. You could drag all the icons you want to select to an uncluttered area of the screen where there are no other icons to interfere, but that takes more time and effort than another alternative.

If you want to select several icons that are scattered about the screen, hold down the Shift key when you select an icon; then point to and select the next icon while holding down the Shift key. If you hold down the Shift key when you select an icon, it will not be deselected when you select another one.

The two procedures allow you to make multiple selections regardless of the location of the icons on the screen. If you make a mistake and want to cancel selections, the standard method of moving the pointer to a clear area of the screen and clicking will work.

Opening and Closing Windows from the File Menu

One of the complaints we have always had against the Apple Macintosh was the decision by Apple to force people to use the mouse. The basic Macintosh, for example, does not even have cursor control keys. You must use the mouse to move the pointer whether that fits your working style and the program you are using or not. We give a higher rating to the Atari ST's GEM desktop environment partially because it has several ways of doing things. Most major desktop tasks can be done with the mouse or with keys on the keyboard.

A good example of the flexibility built into the GEM desktop is the File menu. When you point to the word File on the menu bar, a list of six options will drop down. Several items, such as Format, New Folder, and Show Info have already been discussed.

When a menu drops down from the menu bar, some items will be displayed in dark letters while others are displayed in faint type. The items in dark letters are active and available for use while those printed in faint letters are not currently available. For example, the Format option is only shown in dark letters when a disk drive icon is selected since formatting takes place on one of the disk drives.

The first item in the File menu is Open. If you select a disk drive icon and then select Open in the File menu, the ST will open the window for the disk in that drive. This accomplishes the same thing as double-clicking on the icon.

Selecting Open in the File menu is also the same as double-clicking on a file icon that has been selected. If you Open a program file, that tells the ST you want it to load that program and run it. Selecting Open when a Folder icon is selected opens a window to display the files in that folder.

The File menu also has a Close and Close Window option. These generally accomplish the same thing as clicking on the Close Box of a window. If you select the Close option of the File menu, only the active window will be closed. If there are other windows on the desktop, one of them will become the active window. Selecting Close will also close the most recently opened folder if you have opened a window and then opened a folder in that window. The Close Window option is a bit more drastic. It closes all folders and removes the window from the desktop.

This concludes the chapters on the GEM desktop. Even after devoting two long chapters to the topic, we still have not covered all the intricate details of operating the ST. You are well on your way to mastering this computer though, and more information is provided in the remaining chapters of this book. Much of the information in later chapters is specific to a particular use of the ST such as telecommunications, word processing, or programming in Logo.

5

Word Processing on the Atari ST

Many people think of numbers when they think of computers. Computers certainly are number crunchers; they can perform thousands of calculations a second, and other chapters in this book deal with several types of data processing.

But computers are also for words. This chapter focuses on the role of the computer in helping you create written documents--from a short letter to an important report for work or school. Using a computer to write your own documents and reports is called word processing.

WHY USE A WORD PROCESSOR?

Several surveys indicate that the most common use of personal computers is word processing. A word processing program turns a computer into a tool for creating documents electronically. With a typewriter the words and phrases you write are placed immediately on paper. That's fine if you never make a mistake, never change your mind about a sentence or paragraph, and never have to revise or update a document.

The writing most people do doesn't meet all the criteria just mentioned. Most of us make typing errors. We decide a sentence or paragraph should be rewritten or moved to another part of the document. And we revise reports and papers written last year to bring them up to date.

It is because we make errors, change our minds, and revise documents that word processing on a computer is a greater advance in communication than the typewriter was when it replaced quill pens. A sentence typed on a typewriter is relatively permanent. Simple typing errors are not difficult to correct with correction fluid or correction tape, if you notice them immediately. But what if you type, "This is very, very serious" in the middle of a page and decide later the sentence should be "This is serious"? You could erase the whole sentence, but there will be a blank space in the line because the replacement has fewer letters than the original sentence. Even a simple revision like this requires retyping an entire page! More complex revisions, such as moving a paragraph on page 3 to page 25, may require retyping the entire document!

All the changes mentioned above are easy when you use a word processor. The document you create with a word processing program is a screen image. You can delete words, sentences, even paragraphs and pages, and the program will

automatically rearrange the document so there are no gaps in it. And if you want to add a sentence, a paragraph, or a page or two, word processors have an insert function that will open up space for the insertion. When the document you're writing is a screen image, you can delete, insert, and move material at will.

Writing with a pen or typewriter brings the mechanics of the task to the forefront. For example, when you're writing with a pen, revisions and corrections are left on the page. After a few minutes the page may be littered with the remains of sentences, paragraphs, and thoughts that were deleted, revised, or moved. The same is true, to a lesser extent, when you use a typewriter.

A word-processed document, by contrast, is always clean. When you delete a sentence, it does not remain on the screen with a thick line drawn through it. Paragraphs that were moved are actually in the correct position, and insertions are where they should be instead of written in the margin or on the back of a sheet with an arrow drawn to the insertion point. Word processors remove the refuse of the writing task from your electronic document. Mechanical writing tools do not. That one advantage lets all of us write better and with less effort.

Spelling Checkers. In spite of the advantages word processors offer, some educators do not think their students should be allowed to use them because several types of writing aids are available. Spelling checkers, like *Regent Spell* (Regent Software, $49.95) and *Diction* (Advanced Computer Systems, $14.95), will check the paper or report and identify words that may be spelled incorrectly. Most spelling checker programs let you correct the error on the spot, and many have a built-in dictionary you can use to electronically look up the proper spelling.

Is such a program a crutch that should not be available to children, or adults? Will spelling checkers lead to a generation of writers who spell poorly? We think just the opposite will be the case. When a teacher checks a report and marks words spelled incorrectly, the time between writing the report and receiving feedback on it can be a week or more. By that time the student is working on something else and may not even look carefully at the spelling mistakes. Feedback on errors comes after the teacher has assigned a grade to the report.

With a spelling checker, the feedback on spelling errors is a part of the writing process, and it comes before the paper is evaluated. Students are likely to pay more attention, and they are more likely to learn the correct spelling of words they consistently spell incorrectly.

Spelling checkers do not make corrections automatically. Generally they display a sentence or two from the document and highlight one word that may be misspelled. The person using the spelling checker must decide if the word really is incorrect and type in the correct spelling. That is an outstanding learning experience for the person who needs spelling training. Some even have electronic dictionaries you can use to look up the spelling of a word. Spelling checkers are likely to improve, not degrade, the spelling ability of those who use them.

Grammar Checkers. A similar argument can be made for another type of writing aid--the grammar checker. At the time of this writing there were no grammar checkers available for the ST but that should change quickly as the number of ST computer users grows. Instead of checking for spelling errors, grammar checkers look for common writing errors such as the misuse of similar words like principle and principal, effect and affect. These programs generally list a sentence or two on the screen, point out the possible error, and may even make a suggestion about how the error can be corrected.

I (JW) use a program called *Grammatik*. It is available for many personal computers and does an excellent job of checking for 15 types of common errors. It analyzes writing style at both the word and sentence level, and checks for poor, archaic, and wordy usage, sexist language, improper punctuation, and much more.

When I evaluated reports written in one of my graduate classes, I found over 70% of them could have been substantially improved had the students used a program like Grammatik or *Punctuation + Style*. The same probably holds true for business correspondence and corporate reports.

Although no grammar checkers for the ST were available when this was written that should change as the number of ST computers being used grows.

SELECTING A WORD PROCESSOR

A wide variety of word processing programs for the ST is already available and new ones are coming out frequently. Programs tend to fall into one of two broad categories: GEM-based, menu-oriented programs,or command-oriented programs.

GEM-based programs have the familiar menu bar across the top of the screen and allow you to use the mouse to select the desired option. Menu driven programs eliminate the need to memorize long lists of commands before gaining full use of the program. However, they do have negative points. Once you become familiar with the program's operational characteristics, menus may slow you down. For that reason, menu-driven word processors are preferred if you are an occasional user and likely to forget how the program works between sessions.

Many people begin with a menu-driven program, but most users who rely heavily on their word processors soon move on to command-driven software. Command-driven programs allow you to use one-key or a two-key combination to enter commands. In many programs, the commands are mnemonic. That is, there is some logic to the keys you press to issue commands. For example, Control-S may be the command to Save a document or Control-D the command to Delete.

Several of the GEM-based word processors for the ST will let you move from a menu- driven to a command-driven program without changing software since they offer you the option of entering commands with the mouse or from the keyboard.

Deciding whether you prefer menus or commands will often depend on how often you will be using your word processor. But there are other factors to consider when shopping for a word processor as well. With that in mind, this section describes some of the features you may want to consider when shopping for a word processing program for personal or business use.

Desirable Word Processing Features

Delete Modes. With some programs you must delete material one character at a time. Others allow you to delete words, sentences, paragraphs, and blocks. (Blocks are discussed soon.) The more delete options you have,the more versatile the software. When deleting more than a few words, the program should verify the command. That is, it should check with you before carrying out the command. For example, the program may ask REALLY? Y/N and require you to press Y before deleting a sentence, paragraph, or page.

Insert Modes. This refers to the methods available to add or insert material into an existing document. A destructive insert allows you to replace material by typing over it. Non-destructive insert pushes the old text out of the way as new material is added. Or it creates a blank space where new text can be typed. Both types of insert are desirable. Some programs do not have a destructive insert mode, an inconvenience that sometimes requires you to delete old text before inserting new material.

Block Modes. A block is a section of text you have marked electronically. For example, you might press the control and down arrow keys at the beginning of a section and press them again at the end of that section. The computer can then be told to perform different operations on that block of text. In a full-featured program, blocks of text can be moved, deleted, or copied. Copying a block means you can put the same material in more than one place in the text. Reports often contain many complicated tables that have the same format but different numbers. With block copying, you can type in the table form once, then insert it as many times as needed. All you have to do is fill in the numbers. Most programs allow only one block of text at a time to be marked. A few permit multiple blocks.

Cursor Control. The cursor, generally a solid or blinking rectangle of light, tells you where the next input will appear on the screen. It also tells you what will be affected by a command to insert or delete material. For example, you must move the cursor over a letter to delete it in most programs. Simple word processing programs let you control cursor movement with arrow keys on the keyboard that point in the direction the cursor will move. Movement is generally one character or line at a time. More powerful word processors have ways of moving the cursor a word, sentence, paragraph, page, or block at a time, and to the top and bottom of the screen as well as the beginning and end of a document.

Search and Replace. If you misspell a word throughout a document, does the program have a way of electronically searching for each occurrence and correcting the error? If you are working on a large document, can you find a particular section by telling the computer to search for a word or phrase at the beginning of that section? There are several variations on the search and replace theme:

Search-for-keyword looks for a word you specify and moves the cursor to the point where the word (or phrase) appears.

Replace-once finds the target word and replaces it with a new word you specify.

Selective-replace finds each occurrence of the target word and asks if you want to change it to the new spelling.

Global-replace finds each occurrence of the target word and automatically changes it to the replacement word you specify.

Wild-card search-and-replace allows you to specify a target word with wild-card letters. Telling the program to search for Th?mes, for example, means words like Themes and Thames would be found because the third letter is a wild card. Any letter is acceptable in that position.

Page and Printing Features. When your document is printed, can the number of lines, the number of characters on a line, and the amount of space left at the top and bottom margins be controlled? Will the program work with many different types of printers, or is it limited to only one or two? Will it support underlining, boldfacing, italics, large characters for headings, justified right margins, centered text, sub- and super-scripts, headers and footers, and automatic page numbering? All these features make the program more flexible.

What You See is What You Get (WYSIWYG). Many word processing programs can produce underlined, italicized, and boldface print when used with a printer that supports those options, but quite a few are not capable of displaying print on the screen in those formats. Instead, they require you to add in-line commands to your document.

When you type /U for example, that may tell the program to begin underlining text. You must type another /U when you want to stop underlining. Some programs have the WYSIWYG feature which means you see italics, boldface, underlining, even subscripts and superscripts, on the screen. The ability to view the

text exactly as it will appear on the printed page allows you to make adjustments and cosmetic editing before producing a hard copy on the printer.

Automatic Footnoting and Indexing. In some disciplines, footnotes are placed at the end of an article or chapter. In others, footnotes are placed at the bottom of the page. Most word processors require you to add footnotes manually, a tedious task at best, and a very frustrating one if you must place them at the bottom of the page where they are cited. If you will be writing documents with many footnotes, you may want to look for a word processor that handles this task automatically. A few programs allow you to insert footnote numbers as well as the footnotes themselves into the text. The program then takes over the job of leaving space at the bottom of the page for printing them or saving them for printing at the end of the document or chapter.

Programs with automatic footnoting often handle another job, creation of an index for your paper. You place a special mark beside each term to be included in the index, and when the document is complete, the program automatically generates and prints an index that lists the terms you marked and the pages where they are discussed.

Graphics Imports. A few word processors allow you to create high quality graphics, pictures, and charts with other programs then add those graphics to text files. This feature, which is rare today, is likely to be more common in the future.

Windowing and Multi-tasking. Another increasingly popular feature is the ability to divide the computer display into two or more windows and edit more than one document simultaneously. For example, you might be working on a report and want to include a paragraph from a document you wrote several months ago. Some programs allow you to divide the screen into two separate windows, each of which can display a document. You could load the old document into one window, select the section you want to use in the new document, and then move that block of text from one window to the other.

WORD PROCESSORS FOR THE ST

Recommending specific word processing programs in a book is risky since the field changes quickly. New programs are announced regularly that cost less, do more, and make word processing easier. The following descriptions of programs are intended to give you some idea of the variety that is currently available.

1ST WORD. Since this word processor comes with the ST when you buy it, you surely can't complain about the price! And, for a freebie, it's really a fairly versatile program. It is one of many ST programs which originate in another country. It comes from GST Holdings in Cambridge, England.

1ST WORD is one of those programs which is GEM-based so it has menus but it also allows you to use the function keys across the top of the screen for entering commands. F1, for instance, starts boldface printing. Pressing F1 again stops the boldfacing. In other words, the F1 key "toggles" boldfacing off and on. Other function keys can be used to indicate underlining or italics or to switch between overwrite--destructive insert--and insert--nondestructive insert--modes.

Centering and indenting are accomplished with function keys. You can use the cursor keys on the keyboard or the mouse to move the cursor around in your document. You can delete characters, words, lines, or blocks. This program offers copy, cut and paste, and move options for blocks of text.

This program is designed to be easy to learn and easy to use for the inexperienced and the experienced. In addition to having some very sophisticated features, 1ST WORD has a disk tutorial and a help option available within the pro-

gram which can be set as a safeguard to display any time you enter a command. If you enter the command to Save your document, for example, a dialog box will appear explaining what the Save command does and allow you to cancel the command if that is not what you intended. You can do everything but enter the actual text of your document without touching your computer keyboard if you are a novice at word processing or just prefer to use the mouse. The menu bar is across the top of the screen. And, along the bottom is a series of 10 icons representing the ST's function keys. Clicking on one of these icons produces the same result as pressing the corresponding function key.

One of the unique features of this program is its different type fonts. On the first screen you see when you load 1ST WORD, the setup screen, you will see a box in the center right portion of the desktop. Inside the box are a variety of characters. This is the Font Table which shows characters you may need, or want, to use in your documents but that are not a part of the 256 characters in the regular character set. All you need to do to insert any of these characters into your text is place the cursor at the place where you want the character inserted and use the mouse to select it. Then the desired character will automatically appear in your document at the cursor location.

The Final Word. Perhaps one of the most powerful word processors available for the ST is The Final Word (Mark of the Unicorn, $145.00). It is a conversion of one of the more popular word processors for IBM computers. This program is entirely keyboard-based since, according to the manual, "the great debate about whether it is more or less efficient to do word processing tasks with a rolling rodent is not over yet."

Some companies spend a lot of time, money, and effort converting programs designed for one computer to run on another computer but don't spend equal amounts of time and effort on converting the documentation. Final Word comes with a 337 page Reference Manual, a separate Introduction and Tutorial Manual, and a Quick Reference Card that is longer than the entire manual for some programs. But, there are only four pages of documentation--at the very back of the Reference Manual-- that refer specifically to the ST. Of course, many of the commands are the same as those used on the IBM but even trying to run the tutorial can be a frustrating experience without appropriate documentation.

Final Word is an incredibly complicated word processor but it is probably worth its price and the time you will spend learning to use it if you do a lot of writing. The program supports almost 50 printers. It makes use of a "swap file" to save your text periodically as a means of crash recovery. If the power goes out the text can be recovered by simply running the Recover program on the disk. Of course, any corrections or additions you have made since the last save to the swap file will be lost but this at least minimizes the loss.

Microsoft Word. Another popular word processor originally designed for IBMs and compatibles is *Microsoft Word*. By the time you read this an ST version should be available. *Word* is a very sophisticated yet easy-to-learn program. The disk tutorial which accompanies the tutorial is the best we have ever encountered. It offers a thorough introduction to all aspects of Word for novices and allows the more experienced user to skip those sections which are already familiar.

Word is a combination menu-driven and command-driven program. All features can be accessed through a series of menus which are displayed across the bottom of the screen. But many of the more commonly used features are also accessible with one or two keystroke commands.

Drivers for most popular printers, including laser printers, are supplied with the program. A wide variety of type fonts are available with Word. Such features as automatic page numbering, indexing, outlining and style sheets add to

its versatility. Style sheets allow you to set formats for the layout and margins for pages, placement and spacing of paragraphs, and type font and size, as well as italic or boldface characters, once, and then apply them to your document with simple commands.

Microsoft Word is sophisticated enough for the most complicated word processing task. Yet the tutorial and the help screens available within the program make it one of the more easy to learn and use word processing systems available for the ST.

HippoWord ($89.95). Hippopotamus Software has committed a considerable amount of money and effort to developing software for the ST. This is their full function word processor which is GEM-based and mouse-driven. However, it also has a full range of commands available using the ST keyboard. HippoWord has automatic header and footer, multiple column format, and built-in mail merge capabilities.

We found the variety of type fonts available within the program to be one of its most outstanding features. There are 12 different fonts on the program disk. Hippopotamus also markets a program called *HippoFonts* ($39.95) which can be used with HippoWord to expand the options even more. HippoFonts is the first in a planned series of similar disks. If more are released, the type fonts available for the ST using this word processor may be limitless. To go even further, *HippoPixel* ($39.95) is another program which you can use to create your own custom designed fonts if you want something different from those already available.

Another advanced feature of HippoWord is the search and replace capabilities. Not only can you search for a particular word or phrase and replace each one individually but you can also search and replace globally. You can perform a simple search which will result in a display of all the lines containing the search string. The search function does treat upper and lower case letters differently so you must be careful to enter the word or phrase exactly as it appears in the text. And, if the phrase contains uppercase letters in some places and not in others, you will have to perform a search for each form of the phrase.

The most sophisticated aspect of HippoWord's search and replace function, however, is the use of what it calls "regular expressions." Regular expressions are simply keys which are given special meaning when used in search and replace tasks. You could use a regular expression like [0-9]{3}-[0-9]{4} to search for a telephone number in your text. Although that expression really looks more irregular than regular what it actually does is match a set of three digits followed by a minus sign followed by a set of four digits. The manual which comes with HippoWord is well-written and has a very thorough discussion of these search and replace capabilities.

Other programs from Hippopotamus include *HippoSpell* ($49.95), a 30,000 word spelling checker which can be used with HippoWord and other ST word processors such as *Express*, *ST Writer*, and *GEM Writer*. HippoConcept is an idea processor and will be discussed later in this chapter.

Regent Software Word Processors. *Regent Word* ($49.95) and *Regent Word II* ($99.95) are this software companies entries in the ST word processing market. Word is not GEM-based; Word II is. We ran Regent Word and found it to be a no frills, but very functional, program. It offers search and replace, delete and replace, and block deleting and moving options. The function keys are programmed to carry out the most common commands so all you need to do is press one key on the keyboard to accomplish the task. You can generate headers and footers on each page, center text, set type pitch and font, turn on the underlining feature, and designate super and subscript characters with two key mnemonic commands. Control-

C, for example, centers text and Control-F allows you to define the footer to be printed at the bottom of each page.

Regent Word's unique feature is its ability to send and receive documents with a communications function. Pressing the Alternate and C keys takes you to the communications mode. All you have to do is specify the baud rate to transmit the data and tell Word if you are sending or receiving. Then the document will be transferred over telephone lines. This function requires the use of a modem for communicating the files. You can read more about modems and sending information over telephone lines in the telecommunications chapter of this book.

HabaWriter ($74.95). This program is Haba Systems' GEM-based word processor. Unlike some other programs, it actually shows underlined and bold text on the screen. It has a What You See Is What You Get display. HabaWriter also has windowing capabilities. You can open as many as six different windows, load different documents into each window, and move text from one document to another. The manual for this program contains a Quick Reference Guide and Index which gives step-by-step instructions for mouse and keyboard methods of carrying out commands. This can be helpful if you just need a reminder and don't want to search through the manual itself.

Business Letters (Haba, $49.95). Haba is also marketing a series of programs under the Solutions title. One of these, *Business Letters*, contains 50 prewritten business letters and memos which can be used as-is or edited to meet your particular needs. The letters are divided into eight categories ranging from Employees to Stockholders to Public Officials. One letter recognizes an employee's birthday. Another is a credit application rejection. And still another can be used to design press releases. This type of program is called a "template" because it fits into another program. Templates help you customize a general purpose program for a specific task.

IDEA PROCESSING--A NEW WRITING TOOL

Thus far, we have dealt with programs that assist you in the writing process. Word processors work when you are writing, and supplementary software such as spelling and grammar checkers help identify and correct many errors that otherwise might interfere with communication.

Spelling and grammar checkers do their work after the main writing task has been completed. Another type of software, called "idea processors", "thought processors" or "outline generators" is used before you begin to write.

Paul Bonner introduces idea processors in a *Personal Computing* article. He points out that most uses of personal computers are merely electronic versions of older products. The filing cabinet has become an electronic database; the typewriter is now the word processor; a Rolodex file and calendar evolved into a scheduling program; and the books of a company are now kept with accounting software instead of ledgers. Bonner believes spreadsheets were the first type of software that really was a new tool and not simply a more efficient way of performing existing tasks. Spreadsheets were the first new tools for number manipulation, and he believes that idea processors are the first new manipulation tools for people who deal with words.

Idea processors are used when you are thinking about an idea rather than writing. They come into play when your thoughts are disorganized, or when ideas are not yet mature concepts you can use as guides to writing. This stage, which ultimately produces an outline of what you want to write, is where idea processors work.

Idea processors let you put down your ideas and thoughts in random order, as they occur. This stream-of-consciousness process puts a minimum number of restrictions on you while the ideas are forming. Later, when you are ready to begin organizing your thoughts, the program makes it easy to rearrange your thoughts. It organizes them into a coherent outline of what you want to say.

Thoughts and ideas, which can be as short as a word and as long as several pages, can be arranged by headings and subheadings, sorted, searched for electronically, and arranged. When you are working on main headings, the program can hide all the subheadings since they represent small details that may distract and interfere with work at the big picture level. With the press of a key, however, any or all the subheadings can be brought back into view. The order of headings, and the relationship between headings, can be changed at will, with little more effort than pressing a few keys.

Most of us do not think in logical, outline form. When we are working on a project, ideas come to us in a jumbled, disorganized flow that is difficult to capture in our mind, and even more difficult to write down in some organized form. Idea processors adapt to the way we think, and make it less likely an important point or concept will be lost in the debris as unorganized ideas are shaped into a conceptual whole. In many ways, idea processors deal more with the structure of knowledge and ideas than with the ideas themselves. That is an important skill in many professions.

At present there is only one specific purpose idea processor available for the ST: *HippoConcept* (Hippopotamus Software, $89.95).

PaperClip Elite. This program from Batteries Included is a new generation of a popular word processor for Apple, Commodore, and Atari XE/XL computers. PaperClip Elite combines an idea processor, sophisticated word processor, and a spelling checker that works while you are writing. It also allows for integrating graphics with text and provides windows for cutting and pasting between files.

More idea processors for the ST should be available as this type of software becomes more popular and as the number of ST computer users increases.

6

Business and Professional Applications

Many jobs require gathering, analyzing, processing, synthesizing, and interpreting information. Typically, the people who do those jobs use a computer. A less apparent fact is that all citizens, regardless of their job, also need to process many different types of information.

The federal income tax form, for example, has evolved from a single page into a thick booklet that is difficult to understand. Families making poor tax decisions may pay thousands of dollars more than other families with a similar income and expense pattern. Major purchase decisions such as buying a car or a home, as well as selecting a pension plan or making investments, involve gathering and processing information. All are much more complicated today than ever before.

Using your Atari ST to gather and process information does not guarantee that your financial decisions will lead to early retirement or a life of leisure. But it can shoulder some of the data processing load. Several hundred specialized programs for the ST put it to work on tasks such as managing an investment portfolio, keeping the books of a small business, filling out personal income tax forms, computing statistics for market surveys and research studies, analyzing test item data, and keeping track of family finances. Though such specialized programs are both useful and popular, it would be difficult to mention, much less discuss, all the different types available today. This chapter focuses instead on general-purpose software you can adapt to many special applications.

THE ELECTRONIC SPREADSHEET

Accountants and others who analyze different types of numeric data have used spreadsheets for many years. A spreadsheet is simply a rectangular worksheet with rows and columns of numbers. Each intersection of a column (usually labeled with a letter) and a row (usually labeled with a number) is a **cell.** Cell C4, for example, is where the third column and fourth row intersect.

Cells can contain numbers, such as test scores, text that explains the spreadsheet--"Scores for Test 3"--or formulas that tell the spreadsheet how to calculate with numbers in cells.

A loan amortization table is an example of a simple spreadsheet. It consists of an organized set of numbers that provide information on a specific topic. Salary grids, inventory data, data on an investment portfolio, and a gradebook are also spreadsheets.

Manual spreadsheets, such as the gradebook a teacher uses to keep track of test scores or the columnar pad a construction engineer uses to develop cost estimates and job bids, have two major drawbacks. First, the computations involved in completing the spreadsheet must be done by the user. Computing the average score on a test or the labor and materials cost for a construction job are two examples. With an electronic spreadsheet, the computer handles the computations for you. The simple formula, SUM(B1:B26), for example, tells the computer to add all the numbers stored in cells B1 through B26. If that formula were in cell B27, and the contents of cells B1 through B26 were test scores, cell B27 would contain the sum of those test scores. B28 might contain the formula AVG(B1:B26). That formula calculates the average or mean score on the test. The result would be displayed in cell B28.

Automatic calculation is one of the strengths of electronic spreadsheets. Once the formulas are entered correctly, you do not have to worry about making a calculation error. The computer does the calculations for you. That reduces errors and saves time.

A second disadvantage of manual spreadsheets is their static nature. Suppose you change the grade on a test that has already been entered into the gradebook, or must make room for several new students who join your class half way through the year. With a manual spreadsheet, such changes may call for many recalculations. With an electronic spreadsheet, changing a grade on a test is no problem. The computer will automatically recalculate things like the mean score on the test, and statistics such as the standard deviation and percentile rankings.

Adding new students to an electronic version of a gradebook is also not a problem. Most spreadsheets have an insert command that allows you to add new rows--such as grades for a new student--and columns--for example, scores for an unplanned test or project--without modifying the formulas you wrote. The program can automatically adjust formulas to take care of the additional data. Spreadsheets also have commands that automatically do things like arrange a student roster alphabetically or provide a printout of test scores arranged from highest score to lowest score.

Most occupations involve some manipulation of numeric data. Electronic spreadsheets allow you to do that work more efficiently, and with less likelihood of errors. That brings us to one of the most important advantages of electronic spreadsheets. Since they automatically adjust to changes in the data they contain, you can ask "what if" questions without worrying about the time it takes to compute the answer.

For example, a company can ask, "What if we hire 20 additional employees at Level One on the salary grid and authorize an across-the-board raise of 8%?" If the salary data for the company is in a manual spreadsheet--or worse yet scattered through several spreadsheets--the answer to this question might take days, or weeks, of work. However, if the data is in an electronic spreadsheet, the question can generally be answered in a few minutes.

Thousands of people use their electronic spreadsheets each day to ask "what if" questions. Many of those questions could not be answered without electronic spreadsheets because too much effort is required to obtain the answer.

Today's electronic spreadsheets are used for many applications. They keep track of the cash flow in businesses; project inventory needs based on sales level; generate business plans and financial statements; analyze the data generated in medical experiments; track stock and futures trading patterns; calculate the net present value of investments; analyze buy versus lease options for corporations; and compute the likely value of real estate investments. They do those things and thousands more.

Spreadsheets are versatile because they are general-purpose programs. You can adapt them to many different applications.

Spreadsheets for the ST

Electronic spreadsheet programs, like other applications programs for the ST, can be classified by whether or not they are based on the GEM desktop format. When you are shopping for a spreadsheet your choice may depend, at least in part, on whether you prefer to use the mouse or the keyboard for most of the commands.

GEM-based programs will be more "menu-oriented"; programs which are not GEM-based, "command-oriented." As discussed in the chapter on word processing, menu-oriented programs appeal to infrequent program users because the commands are easily accessible on the menus. Command-oriented programs are usually more appealing to the user who works with the program often enough to remember the commands from use to use instead of having to relearn them each time. The frequent user may find using menus time consuming.

ST Calc (Royal Software, $49.95). This Gem-based spreadsheet includes a built-in drop-down calculator. ST Calc has menus available but also makes full use of the ST's ten function keys. One of ST Calc's more unique features is the move function which allows you to use the mouse to move information from one cell to another within the spreadsheet.

A-Calc (Antic Software, $59.95). With this Gem-based program you can use the ST's windowing capabilities to look at two different parts of your spreadsheet at the same time. And, if you have ever worked with a spreadsheet that is too large to fit in the screen display, you know that feature can be a definite advantage. Antic has released several programs which they call the A-Series. The series includes a database, report generator, telecommunications program and a couple of utilities programs. A-Calc allows you to transfer data to and from any of the programs in this series.

SwiftCalc. Timeworks' popular Commodore 64 and 128 spreadsheet, SwiftCalc, is reportedly being converted to run on the ST. The program is a very good spreadsheet which can handle 250 rows and 250 columns. That is a total of 62,500 cells, more than enough for all but the most complex applications. SwiftCalc comes with an easy-to-understand manual and has help messages you can call up on the screen for quick reference.

This program is a solid, basic spreadsheet that is excellent for light- and medium-duty work. You do not have to learn many commands because it can display menus on the screen. You then select options from the appropriate menu. The program also has a number of convenience features. For example, if you must enter labels for your spreadsheet such as the months of the year or days of the week, you can type in the first one (e.g., January, 1987), and SwiftCalc will take it from there. A command lets you move the cursor to the next cell and the program will type in February, 1987. That automatic labelling can continue for as long as you like as long as the days, months, or years are in consecutive sequence. This fea-

ture saves steps when you must enter sequential dates, and can also be used for headings such as Region 1, Region 2, and so on.

Spreadsheet Templates. A disadvantage of spreadsheets is the fact they must be adapted. Electronic spreadsheets are available for the ST, but they are not ready to do anything for you. You must create your own customized spreadsheet by entering the formulas, setting up columns and rows for data entry, and organizing the spreadsheet for easy use. This task is somewhat like writing a program in BASIC or Pascal, but it is generally easier because the spreadsheet program has been designed to do a specific type of work--analyze and manipulate numeric data--and it has many special commands to do that work.

If the use you have in mind for a spreadsheet is relatively common, there is an easy way to use non-customized spreadsheets. Several publishers sell "templates" for popular programs. Templates are customized spreadsheet forms that convert the general-purpose spreadsheet into a tool that performs a specific job. *Help Calc ST* (Royal Software, $24.95) is a disk of templates for ST Calc and another spreadsheet program for the ST which will be discussed later in this chapter, VIP Professional. These templates are designed to help with such tasks as figuring depreciation and loan amortization schedules.

DATABASE MANAGEMENT SOFTWARE

The term "electronic file cabinet" is often used as a literary substitute for more technical terms like "file management system (FMS)" and "database management system (DBMS)." Simple programs that do things like keep track of names and addresses, or the records in your jazz collection are called file management systems (FMS). FMS programs are limited, but useful. More sophisticated programs are generally referred to as Database Management Systems (DBMS). All of these terms refer to an organized means of storing information.

If you store your information in a file cabinet, you may spend a great deal of time maintaining the "organized means of storing information." And it may not always be that organized.

If you use a computer to store information, it is still necessary to set up the format of the electronic files yourself. The file management or database management system will do much of the work required to add new information to the file and retrieve specific pieces of information from it.

You may not be the comptroller of a multimillion-dollar corporation, but files may also play a crucial role in your work. A computerized file management system will probably save you time and provide easier, better access to the information in your files. Consider this hypothetical example: You are the president of a state professional association and have the responsibility of keeping track of the officers and membership rolls of local chapters. Those records are a database. That database may be in file folders, arranged alphabetically by local chapter name, or in four drawers of a filling cabinet. No matter where the information is stored, it is a database.

If you want to find the names of the membership secretary of each local chapter, you should be able to go to the cabinet and look through all the folders until you come across each name. Your four-drawer database may serve as an important source of information but maintaining it may take quite a bit of time. You'll often find that filing the information by chapter means you'll have to spend hours

searching through the entire database, looking through each of those folders to find the information you need.

Another option is to computerize the information. This is a relatively complicated database, so you would need a sophisticated database management system (DBMS).

There are major advantages to taking the database out of your file cabinet and putting it on computer. First, the time required to update and maintain the file is drastically reduced. Often it can turn a full-time job into a part-time task. Second, retrieving complex data takes 1/10 to 1/100 the time it would take by hand. If you use a file cabinet, you must file material under one main heading. In an electronic database you can file one piece of information under 15 or 20 different identifiers and then retrieve that information using any of those identifiers or a combination of identifiers. Third, having the data in a computerized database lets you get information in ways and forms that would be too time-consuming or impossible if the database is in a file cabinet.

Basics of Database Management Systems

Although there are over 100 books currently in print that discuss database management software, many of them deal with large, mainframe applications. A sizable number of the rest are for programmers and systems analysts involved in the development of database management systems. The information you need for a decent database management program isn't the same type a programmer needs to write database management software. This section is not a comprehensive overview of DBMS jargon, but it does cover a few basic concepts and information that may help you when you go shopping for a database.

Database Files and Functions. A database is an organized body of information. The term is often used in a very broad sense, but it also refers specifically to the entire body of information kept in computerized files. In a computerized information management system, you will need a program, or set of programs, to manage your database.

Regardless of the term used to describe the program, the information within it will be divided into files. Each file is organized into units of information called fields. In our example, there would be fields for data such as the name of the local chapter, the officers, a mailing address, phone number, and list of active members.

A database will perform four basic functions. First, you must be able to input your data into the database the program will manage. DBMS programs vary considerably in the number and type of fields they will accept. A program, for example, that permits no more than ten fields per record would not be suitable for the organization's file because there will be more than ten fields of information in each record. A program that lets you have as many fields as you want would be nice, but what if the program sets aside space for 500 characters of information in each field? That would waste space in the computer's memory and on the disk where the data is stored because some of your fields will need no more than three or four characters.

Once the data is entered, the program must be able to take the data and sort, search, and organize it in different ways. For example, you may want the data you input on local chapters reorganized so that it is arranged according to regions of the state. Or, you might want to search each record for the name of the local president.

A DBMS program processes data. For example, it should be able to take information on dues received from members of each chapter and perform the necessary calculations to produce aggregate data on the financial status of the organization.

The database management program must also be able to produce a variety of reports. Reports, or output, are the harvest you receive from your database. For example, you may want the system to produce a mailing list on gummed labels for a newsletter you want to send to a selected set of addresses in your database. That task requires the system to select a few pieces of information (the name and address) on a subset of all the individuals in your database--such as the corresponding secretary from each chapter across the state--and to print out that information on special mailing labels. You might get that report from your database before noon and then ask it to generate the mailing labels before the afternoon mail goes out. To be most useful, the DBMS must be able to generate many different types of reports.

Shopping for Database Software

Some aspects of selecting a DBMS program have already been mentioned. This section offers a few more suggestions.

Be Sure The Software Is Compatible With Your System. Many software vendors are prone to reflexively answer "Yes" to the question, "Will this program run on my computer?" Often the answer is based less on fact than on the salesperson's desire to make a sale.

Consider Ease of Use Versus Sophistication. You will probably want to shop for a system that does the work but is not unnecessarily difficult to use because of complicated procedures you don't need. If an easy-to-use program will do your job, it may be better for you than a more complex program.

We should also mention a problem related to evaluating the sophistication and ease of use of programs. You are trying to use the computer to accomplish a goal, but you may not have years of training and experience with computers and programs. A program described as easy to use in a technical magazine like *Byte* may be considered very difficult in reviews published in magazines like *Family Computing*. Reviews in both magazines may be correct but a difficult program for the applications user may be easy to the programmer.

Look for Specific Features. Most applications put more emphasis on one feature than another. For example, some uses require the DBMS to frequently search and process data. In that case a program that does searches and sorts quickly on most, or all, of the fields is desirable. Other uses place more emphasis on the ability to use may fields in each record, or fields of variable lengths.

A desirable feature not mentioned already is the ability to create custom input screens. Programs like *H & D Base* ($99.95) from Mirage Concepts let you create customized screens that facilitate input. For example, you could create the video equivalent of a commonly used form. When you work with data from that form, the screen will look like the form. When you enter data in the first blank on the form the cursor will jump to the next blank where data should be entered. Using customized screen forms makes data entry easier.

A final feature to evaluate carefully is the versatility of the program's report generating system. The more freedom you have to design the report format, the more likely you are to be able to get exactly what you want. H & D Base is a GEM-based program which can generate a wide range of reports. Information

from more than one of the databases you have created can be combined into one report.

Database Software for the ST

There are already a wide variety of database programs available for the ST.

ST-File (Quantum Microsystems, Inc., $39.95) is a "relational" database. That means data in one field can be related to that in another field for computation. You can even include graphics in your ST-File records.

Regent Base (Regent Software, $99.95) is a GEM-based database, and a relational database. That means windows, menus, icons, and mouse buttons are available to help make it easier to use. Regent Base allows sorting on as many as three fields at one time to reorganize information within the database.

With this program, you can design your own customized reports which can include features such as multiple type styles, page numbers, varied line spacing, and bold or underlined type. Regent Base has fifteen built-in printer drivers so you should have no trouble getting it to work with most popular printers.

Templates for databases, like those for spreadsheets, help take a general purpose program and make it more appropriate for a specific task. Regent Base comes with one template for balancing your checkbook and another which lets you take a database of names and addresses and merge them with a form letter created on Regent's word processor to create mailing lists and labels.

Business Template I ($49.95) is the first in a series of template disks designed to be used with Regent Base. It includes templates for common business financial record-keeping tasks such as accounts receivable and payable, invoicing, and general ledger.

Data Manager. When Susan Tanoa reviewed the Commodore version of this program for *RUN* magazine, she concluded, "Timeworks has succeeded in making a program that is both powerful and easy to use." We agree.

Data Manager has all the features you expect in a file management program. It is not as easy to learn as SwiftCalc, but it offers a generous amount of flexibility in the way you design fields and generate reports. One feature we like is the ability to delete or add a field to records after the file is created. Many databases do not permit that, and we always seem to think of one or two more fields that should be added. Data Manager has several options for finding and organizing information in the file and can generate simple graphs using the data in a file. Data Manager is not a substitute for highly sophisticated programs, but it is far easier to use and will do everything many users require.

DB-ONE (Oxxi, Inc., $99.00) is designed with both experienced and inexperienced users in mind. It includes a tutorial which takes you through the steps in creating your own database records, and sample programs for business and home applications such as keeping credit card records and producing mailing labels. You can even use one of the sample programs to set up an inventory of your business products or your household items.

Experienced users will also find DB-ONE offers some powerful options like defining the input screen and transferring files from dBase II--a popular database for the IBM PC. Like Data Manager, DB-ONE allows you to change the format after your record has been created. It also generates a wide variety of reports.

Zoomracks (Quickview Systems, $79.95) is a unique database con-

cept. The files are displayed as a series of cards in a rack much like the rack you might see next to a time clock. It offers a high degree of flexibility in designing fields and allows you to insert or delete fields after the file is created. Maximum field size of 250 lines of 80 characters makes creating form letters possible since you could simply put the text of your letter in one of the fields. Report generation is nonexistent other than form letters, business forms--such as invoices, and mailing labels. Knowing database jargon will be of little help in learning to use this program since terms such as file, record, and field are replaced by rack, card, and fieldscroll. Zoomracks is menu driven and seems to be easy to learn. The rack concept does seem to make the database concept easier to visualize.

Integrated Programs

The programs discussed so far in this chapter are stand-alone programs. That type of software is very popular, but another type, the integrated program, is growing in popularity. In fact, as integrated product for the Apple II series of computers, Appleworks, topped several best seller lists in 1985.

Integrated software is really a set of programs which usually includes an electronic spreadsheet in addition to programs to do other types of work. One of the first integrated programs, and still the most popular, is *Lotus 1-2-3*. The 1-2-3 program, which runs on the IBM PC, includes an electronic spreadsheet, a database manager, and a word processor, as well as graphics and telecommunications software.

Many people prefer integrated programs because they can learn to use one program and do most of their business and professional computing work with it. There are advantages since programs developed by different companies do not use the same commands and may even require conflicting or opposing patterns of use.

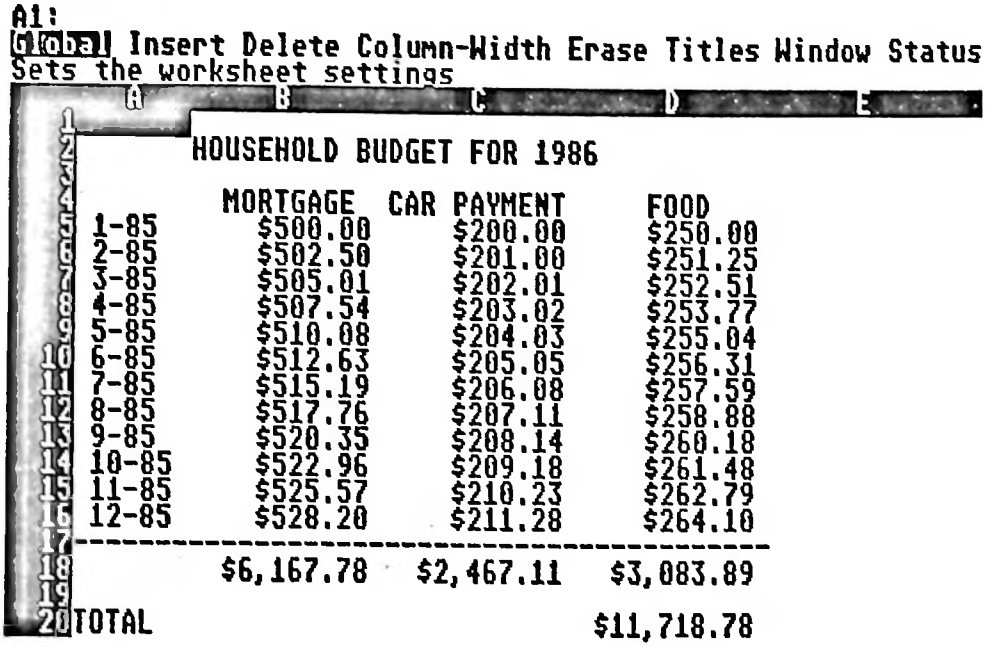


Figure 6.1-A personal budget using VIP Professional Spreadsheet.

VIP Professional. VIP Technologies has introduced a program which is the ST version of 1-2-3. The program, *VIP Professional* ($179.95), uses the same commands and functions 1-2-3 uses. If you have worked with 1-2-3, you won't have to learn a new set of commands. Some conversion is necessary but 1-2-3 files can be loaded into, and used with, Professional. I (GB) was amazed at how easy it was to get into this program. If you have never worked with an electronic spreadsheet or database at all before, VIP Professional has a good tutorial in the manual which takes you through all the steps in creating, editing, and saving a spreadsheet. The concepts and commands are the same for the database and spreadsheet. So, if you complete the tutorial you will learn about creating and editing databases at the same time.

VIP Professional is not copy-protected. The manual recommends that you make a back-up for loading the program before you do anything else. However, you cannot run the program without the original disk because it is the "key disk."

When Professional loads, the screen displays a prompt telling you to insert the key disk and press RETURN. If you do not follow those instructions, the worksheet display never appears. This technique is used by some companies to keep users from copying and distributing software programs illegally. You can copy VIP Professional as many times as you wish, but you still have to use the master disk in order to access the program.

We used a word processing program with a similar protection scheme and found it inconvenient. We had a computer at home and another one at the office. It seemed the key was always at home when it was needed at the office, and vice versa.

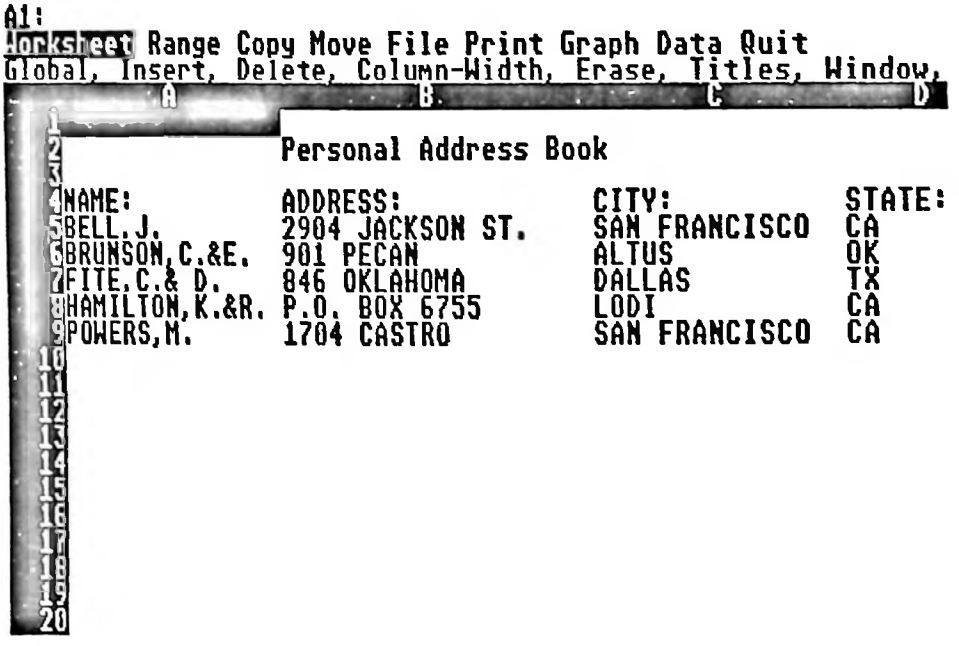


Figure 6.2-Using the VIP Professional Database as an address book.

VIP has a solution for that problem. They provide a Backup Disk Order Form you can use to order another master disk. For $15, if you send in your prod-

uct registration card, you may purchase one backup disk. The company maintains registration records and will provide only one backup disk for each registration number. The maximum size of a Professional worksheet is 8192 rows and 256 columns. That is more than two million available cells--more than enough to handle even fairly complex worksheets.

Many programs which are advertised as integrated software are merely separate programs which can interchange files. VIP Professional *is* an integrated program. The environment is exactly the same whether you are working with a database file or a spreadsheet file. You only have to learn one way of moving around the screen and entering data. Of course, the two applications perform different functions: spreadsheets deal with numbers, and databases with information organization.

Another version of VIP is available. *VIP Light* ($99.95) is less expensive than the Professional version but it includes only spreadsheet functions.

HomePak. Another company, Batteries Included, has an integrated package of software called *HomePak* ($69.95). It contains a word processor, file management program, and telecommunications program. HomePak is a light-duty program that is excellent for home use.

Other Business Software

Some business applications programs fall into categories other than electronic spreadsheets, databases, and integrated software. We will discuss a few of these in this section. There are more programs described in other chapters which, though not exclusively for business, could certainly fit in well in many offices.

Most businesses have lots of documents to produce. That is why word processing is another one of the most widely-used applications. You may want to read the chapter on word processing for information on that subject and on the specific programs available for the ST, two of which provide many types of ready-made business communications. If you use graphs and illustrations in your business communications, the Arts and Crafts chapter discusses some graphics programs which may interest you.

Batteries Included's Integral Solutions. *The Isgur Portfolio System* ($199.95) is one of the Integral Solutions series. *PC Magazine* describes this program as a "capable package appropriate for professional and individual investors." The Isgur Portfolio helps you keep track of all your investments, displays annual gains or losses, allows you to try "what if" situations to determine what effect price variations would have on your gains--or losses, and will even send you a message to let you know what is happening--for instance, if you tell the program to alert you when your Atari stock reaches 150.

Another program in the Integral Solutions series, *I*S Talk*, is included in the Isgur Portfolio. You can use this telecommunications program to get in touch with your investment counselor and buy or sell part of your securities. Talk is set to receive market quotes from Dow Jones, CompuServe, The Source, and InfoGlobe, so your portfolio is automatically updated.

BOS Programs. BOS (Business Operating System) is a popular operating system in England. Atari has reportedly purchased the American rights to BOS and is considering packaging it with the ST. When you purchase your ST, the operating system which is included--either on disk or in ROM--is called TOS. You must have it to run programs on your computer. But, BOS could be used *instead of TOS* to run a wide variety of programs from a company called BOS National. Word processing, database, spreadsheet, accounts receivable/payable, and general ledger programs are some examples of the programs available. These programs are somewhat expensive--$500 and up--and are obviously intended for the business market.

If Atari does include BOS with the ST, it is likely that other companies which already produce BOS software in England will find it a lucrative venture to expand their market to the United States. That would open an entirely new software world to ST owners.

7

Telecommunications: A Window on the World

Many people are aware of the recreational and educational potential of personal computers. Their usefulness in the office is also well known. However, the topic of this chapter, telecommunications, is not as familiar to many owners. Telecomputing, or telecommunications, involves connecting your computer to other computers over standard telephone lines. Your machine becomes a link to thousands of other computers programmed to provide various services to those who call.

The ST can become an electronic gateway to many types of information, and that is increasingly cited as a reason for purchasing a computer. Information is no longer a slow-moving, gradually changing stream. Today the flow of information is a torrent, and it flows by us so quickly that we cannot hope to capture more than a tiny percentage of it for our own use.

There is so much new information that we need some sort of intellectual lever that will allow us to take our intellectual talents and leverage them to obtain more information processing power. Fortunately, there is just such an intellectual lever -- the personal computer. In the last ten years, hundreds of information and communications services have been developed specifically for the personal computer owner. This chapter introduces and explains many of those services and discusses the peripherals and software needed for telecommunications.

THE COMMUNICATING COMPUTER

Using the computer as an electronic communications system is probably one of the least understood personal computer applications, but it may be this use that introduces the largest number of people to computers. It may, in fact, have an impact on our way of life that parallels the impact of radio, television, and the telephone.

Societies have always been concerned with information and with communication. Not only do we want to know, we want to know quickly. The original long distance Olympic runners, for example, were honored for their speed partially because they could carry messages relatively quickly. Clipper ships, the telegraph, the Pony Express, transatlantic cables, and communications satellites are familiar

examples of advances that helped us communicate better and faster. Each ushered in a new era of information and communications.

Today we are entering another new era of communication that may change the ways we get and use information and the ways we communicate with each other. That new era, in fact, has already arrived for many. While a farmer in Oklahoma is checking markets and weather forecasts on an electronic information service, a stockbroker on the California coast may be reading stock prices on the screen of a personal computer in his den, and a precious metals trader in Denver may be using her computer to get the Zurich, London, and New York gold and silver prices. The farmer, the stockbroker, and the commodity trader all use the computer for three reasons. First, they get the information they need quickly. Second, they save time because the computer does much of the required clerical work. Press a few keys and the screen fills with information.

A third benefit is convenience. In many homes and offices, computers are used to check on airline reservations, pay bills, read movie reviews, get detailed weather forecasts, communicate with others who have similar interests, and search for information on a topic of interest. Today you can even go shopping by sitting down at your computer.

WHAT YOU NEED FOR TELECOMMUNICATIONS

To use the ST for telecommunications, you will need a modem and telecommunications software. A modem--short for modulator-demodulator--takes the signals from your computer and converts them into tones that can be reliably transmitted over phone lines. It also converts tones transmitted to your computer into signals the ST can process.

Modems

Most standard modems plug into the RS-232 serial port on a computer. All models of the ST have just such a port, and that means any of the several hundred different models of standard modems can be used with the ST. Modems from companies such as Anchor Automation, Hayes, Avatex, and many others will work on the ST. For example, several companies sell a "300-BPS, auto-answer/auto dial modem" for less than $100 that works with touch tone and older phones.

What does all that jargon mean? 300 BPS refers to how fast the modem can transmit and receive data. A 300-BPS modem transmits 300 bits of data a second.

That was fast a few years ago, but today 1200-BPS modems are quickly becoming the standard, and 2400-BPS modems are already available. Most of the modems that sell for less than $50 today are 300-BPS models, but by the end of 1987, 1200-BPS models will be more popular, and several models should be available for less than $100. (The term baud is often used in place of BPS in advertising and documentation. In general usage, BPS and BAUD mean the same thing when used to describe the speed of a modem.)

An auto-answer modem can answer the phone and connect the computer to the phone line. That feature allows you to call your computer from a remote location and transmit or receive data from it.

An auto-dial modem can automatically dial numbers for you. That feature frees you from the task of dialing the number yourself and then connecting the mo-

dem to the phone line when the computer on the other end answers. With appropriate software, an auto-dial modem can even redial busy numbers until it gets through.

When this chapter was written, several 300-BPS, auto-answer, auto-dial modems were available for less than $75, and one 1200-BPS, auto-answer, auto-dial modem made by Avatex was selling for under $100 from a few computer mail order suppliers. In most towns with 100,000 or larger populations, modems will be available through computer stores, discount stores such as Target, BEST, Federated, and K-Mart, at Radio Shack, and even at Sears. Prices may vary drastically from store to store even when you are looking at models with similar features, so it is a good idea to shop around before making a purchase.

In addition, modems are generally reliable products that work for years without problem, and prices are sometimes even lower if you buy one from a mail-order supplier. If we were buying a modem for our ST today, we would look for a 1200-BPS, auto-answer, auto-dial model that is "Hayes-compatible." Hayes compatibility is discussed later. In September 1986, we could pay as much as $300 for such a modem (a genuine Hayes model purchased at a local computer store) and as little as $79 (for an off-brand modem purchased from a mail-order discounter). You will need an external modem instead of an internal model that was made to be installed inside a computer. Internal modems are available for computers like the Apple IIe and IBM PC that have expansion slots. The ST has no expansion slots, and therefore must use an external modem. It is connected to the computer through a standard serial cable.

When you buy a modem, be sure to ask whether the cable is included with the modem. In most cases, it is not. Standard serial cables can be purchased for as little as $6.96 and as much as $39. The price depends on the length of the cable, the quality, and the source. We have used some very inexpensive cables and never had a problem.

To begin a telecommunications session, you plug the modem into the computer and the telephone line, and then load the telecommunications software into the computer from a disk. There are several inexpensive telecommunications programs for the ST, and some are discussed later in this chapter. Most ST telecommunications can, for example, download data from the remote computer and store it in a file on a disk. Some bare bones programs do not, however, support an error correction protocol. Several telecommunications services have hundreds of programs you can download to your ST and run.

When you receive a program file from a remote computer by phone--called downloading--one error in the thousands of bits of data transmitted can mean the program won't run properly. Since many telephone lines are relatively "noisy," many telecommunications programs employ special error correction protocols that check each byte of data transmitted. If an error is detected, the program asks the remote computer to retransmit the data.

The most popular error correction protocol today is *XMODEM*--sometimes called *MODEM7*--but there are many others, including one called *Kermit* that is used by many universities that provide telecommunications access to their large, mainframe computers.

Regardless of the modem you use, you will probably want software that supports one or more of the popular error correction protocols. Fortunately, several ST telecommunications programs that work with standard modems do support error correction protocols.

We suggested earlier that you restrict your modem shopping to models that are "Hayes-compatible." D.C. Hayes is the dominant manufacturer of modems today and its modems understand a comprehensive set of commands.

Hayes commands begin with AT, which is short for attention. For example if you send the command ATDT7466187 to a Hayes-compatible modem, it knows a command is coming when it receives the AT.

DT tells the modem to dial a number using tones. DP would tell it to use pulses instead of tones. The modem will then dial the number it receives. The advanced features of many telecommunications programs can be used only if you have a modem that understands the commands used by Hayes modems. Other telecommunications programs are set up to work with Hayes-compatible modems, but you can, with effort, change the default settings for things like the command that tells the modem to dial a number to make it work with auto-dial modems that do not use the Hayes command set.

ST Telecommunications Software

When you shop for telecommunications software, look for programs that support popular error correction protocols such as XMODEM and Kermit, and ones that handle both standard ASCII text files as well as files that contain programs or data in Atari's expanded version of the ASCII code. The software should also have the ability to upload and download files -- send and receive data it stores on a disk. You may also need a program that lets the computer emulate the operation of popular terminals such as the VT-100. Several of the information utilities discussed later in this chapter have public domain communications programs for the ST you can download and use without cost.

Mi-Term (MichTron, $50). One popular telecommunications program for the ST is Mi-Term. It operates within the GEM desktop format which allows it to use drop down menus and icons. Mi-Term is also available in versions for the IBM PC and compatibles. However, since GEM is not the standard operating system for the IBM PC, you must buy GEM as well as Mi-Term.

Mi-Term has a number of advanced features. It can, for example, automatically "logon" when you tell it to dial the number of an information utility such as CompuServe. Most electronic information services require you to type your account number and a password each time you use the service. That is referred to as logging on. With this package that task can be handled by Mi-Term. You can even program Mi-Term to transmit frequently used commands such as the instructions that let you check for messages sent to you through the utility. The program also supports XMODEM and DFT (Direct File Transfer) error correction protocols for uploading and downloading data, it will output incoming data to a printer as well as the screen, and it can operate at nine speeds from 110 BPS to 9600 (if your modem can also operate at that speed).

DFT (MichTron, $49.95). Another interesting telecommunications program from MichTron is DFT which is short for Direct File Transfer. This package comes with two disks, one for the Atari ST and another for an IBM PC or PC-compatible. DFT automates the process of transferring data between an ST and an IBM PC or similar computer. All you need to transfer files of data between the ST and a PC is DFT and a null-modem cable. Null-modem cables are available at most computer stores and at Radio Shack, but the manual for DFT includes a diagram that tells you how to make a null-modem cable yourself. DFT lets you connect the two computers to each other directly through the null-modem cable if they are in the

same room. If the ST and PC are not in the same area, you can transfer files by phone if each computer has a modem.

PC/InterComm (Mark of the Unicorn, $124). Digital Equipment Corporation's VT102 intelligent terminal is used by thousands of people to communicate with large computers such as the VAX series made by Digital Equipment and models made by many other manufacturers that run the UNIX operating system. The VT100 is not inexpensive, but it is popular because many large computers are set up to communicate correctly with the VT100. With PC/InterComm your Atari ST can do double duty by emulating a VT100 terminal. This "terminal emulator" software turns the ST into a sophisticated terminal which a VAX computer thinks is a VT102 terminal. The cost of the software and a well-equipped ST is much less than the cost of a VT100 terminal, and you get a very powerful computer as well.

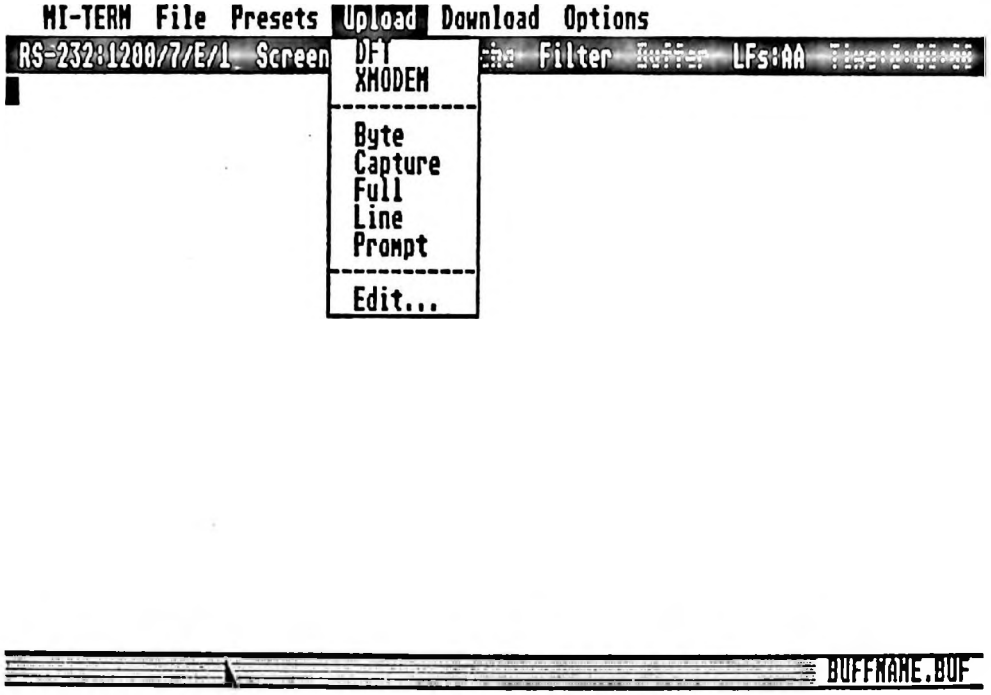


Figure 7.1-The Mi-Term Screen and Upload Menu.

PC/Intercom also has file downloading and uploading capability using the Kermit and XMODEM protocols, auto-dial and auto-logon, and it has features for customizing the keyboard so long instruction sequences can be issued by striking one or two keys. Because of the similarities between Digital Equipment's VT100, VT102, and VT52 terminals, an ST running PC/InterCom can emulate all three. If you need access to a large computer that works with the Hewlett Packard 2392A terminal, a terminal emulation program named *ST-2392* (Coincidence Technical Software, $169) lets the ST emulate this $1200 terminal.

ST-TERM (Commnet Systems,429.95). This is one of the least expensive ST telecommunications programs, but the low price is not a reflection of the value or usability of the program. Developed by Matthew Singer, ST-TERM is a terminal emulation program that turns the ST into an intelligent terminal. It emulates the

VT52 terminal made by Digital Equipment Corporation as well as other terminals that use the ANSI format. It can deal with standard ASCII text files as well as Atascii data (Atari's version of the American Standard Code for Information Interchange). Atascii is an expanded version of the ASCII code that includes codes for graphics characters as well as regular text. ST-Term can upload and download files using XMODEM, Kermit, and several other protocols. The program has an auto-dial feature and it can store the phone numbers and communications parameters of up to 400 different services and automatically dial the number and connect to the service. The program will repeatedly redial a number that is busy until the line is free (called "Attack dialing" in the manual), a feature that was used by a friend of ours to get through to the reservation desk of a tennis court that always seemed to be busy.

ST-TERM can even handle alternative long distance services such as MCI and Sprint that require you to dial a local number before the long distance number. It will also keep track of the time you spend on a long distance connect and compute the cost of the call if you like. Few programs provide so much for such a low price.

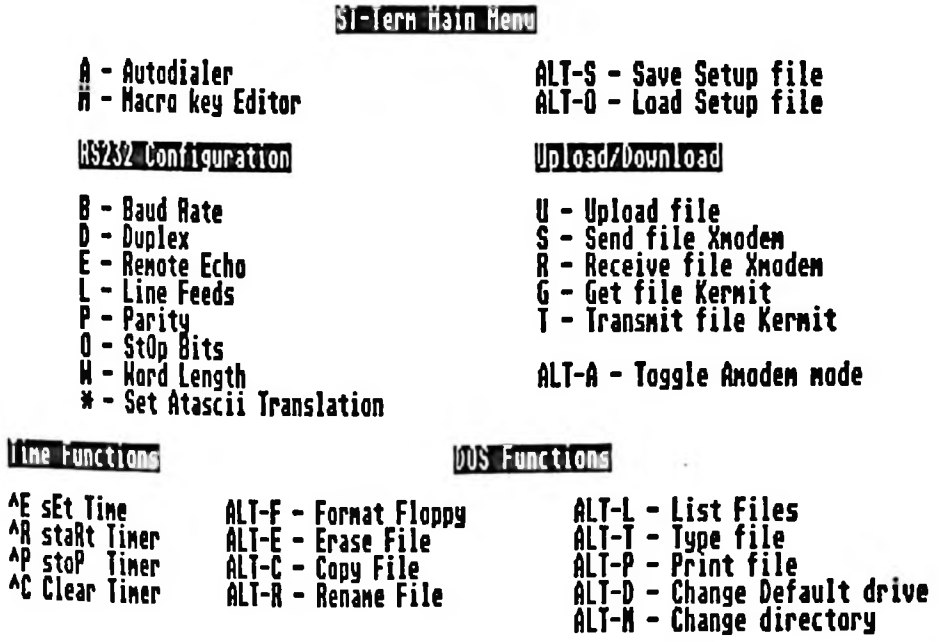


Figure 7.2-The ST-TERM Main Menu Screen.

MAJOR COMMUNICATIONS USES

Nine major uses come to mind when we think of the computer as a communications device: information management, computer bulletin boards, electronic mail, electronic networking, electronic publishing, consumer services, financial services, downloading programs, and computing.

Information Management

If futurists are correct, our society is in the middle of a transition that will have a tremendous impact on our lives. The transition involves a shift from a society founded on an industrial base to an information-oriented society. Finding, selecting, and using information has become a life's work for many people today and a new term-- knowledge worker--has been coined to describe this new working class. Today's best investment, most promising business or profession, or even the most attractive retirement plan may be undesirable or unsuitable in a short time because of changing conditions. Information ages quickly today, and we are all in need of fresh information to make sound decisions.

If you need the current price of Exxon stock, high school football scores from North Carolina, or the time of the next plane to Paris, you can get that, and much more, from one of the many electronic information utilities that are as near as the keyboard of your ST.

Information Utilities. Personal computer owners have access to several general purpose information utilities that offer all sorts of electronic information services. Both *The Source* and *CompuServe*, the most popular general purpose information services, have extensive data banks that can be searched electronically from the keyboard of your computer. There are also hundreds of specialized database services that can be accessed and used by anyone with a personal computer. In a matter of minutes, a physician can find articles about the side effects of a new drug. A farmer can read the current predictions on the wheat crop in Russia. A family can find product reviews of popular video cameras, and a teacher can locate educational programs for junior high school computer classes.

All this information can be found in the many electronic information services in operation today. The information is transmitted to your home or office over the phone line and displayed on the screen of your computer. Information in electronic information services is organized into databases that contain millions of pieces of information relevant to a particular topic. Some databases, for example, have information on the performance of corporations in North America. Others have data on software for personal computers, world news, grants and awards for researchers, programs for Atari computers, and a thousand other topics.

Some writers predict these new computerized information utilities will replace traditional sources of news and information. They believe, for example, the days of newspapers and television news reports are limited. We doubt that. Newspapers are good sources of general information professionally packaged and conveniently presented. Electronic information utilities are likely to be added to our existing sources of information, rather than replace them.

There are several reasons. First, information utilities are not that easy to access. A typical paper costs 25 cents and requires no special equipment to use. Information utilities, on the other hand, require a computer, a modem, and appropriate software that converts the computer into a communications system. In addition, most information utilities require you to pay a fee when you sign up for the service, and they charge for each minute you are connected to the service. You may also find it necessary to learn a complicated set of commands before you can effectively use an information service.

In their present state, computerized information services cost more to access, and have a higher learning overhead than traditional information sources such as magazines, books, and television. They are not viable sources of the same information you get in a newscast or in the paper.

On the other hand, these services are often the preferred method of

searching for special information not normally available in the local paper or on the six o'clock news. Today there are hundreds of databases with billions of pieces of information. There are special databases for lawyers, physicians, psychologists, engineers, home economics teachers, and many other professions. You can use your computer to look up federal court cases on sex discrimination, sources of financial aid for college, or rental apartments in Bermuda. That, and more, is available right now.

Computer Bulletin Board Systems

There are hundreds of local computer bulletin board systems (CBBS) in operation today. You use them by dialing a local phone number that connects your computer to the computer used by the bulletin board. These CBBS are run by people called system operators, or SYSOPS. CBBS may be sponsored by a local computer club, an amateur radio group, or special interest groups such as the local organization for handicapped citizens, by a school, or by a business such as a computer store. Members of the sponsoring organization (and in many instances anyone who knows the telephone number) can communicate with the CBBS computer.

A computer club is a typical sponsor of a CBBS. The club pays for a phone line and connects their computer to it. The club computer answers the phone when you call and transmits a welcome message that is displayed on the screen of your computer. If it is a closed bulletin board, you may then be asked for a password or an identification number which, when typed, gives you access to the features of the bulletin board. If it is an open access bulletin board, you may get a menu of services on your screen.

The menu may have options such as reading the latest issue of the local computer newsletter, browsing through want ads that offer to sell (or buy) used equipment, or looking at comments from other users about new products or common problems. The options available on a local bulletin board will vary from system to system. Some have only one or two services. Some are really special interest boards (e.g., for IBM PC owners who use their computers in business applications). Others seem to be small organizations in themselves. There may be programs you can download from the bulletin board into the memory of your computer, bibliographies and directories of articles you can search electronically for information on a particular topic, a newsletter, and advertisements from companies offering services and products at discounted prices.

Many CBBS are sponsored by local Atari owner's groups and can be used by anyone interested in the latest information and tips on how to use their computer more effectively.

If you would like information on the free computer bulletin board systems in the United States, you can get up-to-date phone numbers and descriptions of services by dialing (213) 881-6880. That is the number of a computer bulletin board run by *Novation*, a modem manufacturer. To use the Novation bulletin board you need a computer and a compatible modem properly connected to the computer and the telephone line. When you dial the number and get a tone, press the RETURN key on your computer. The CBBS should respond with a message to LOGON PLEASE.

Most systems have a "sign on" procedure that involves typing in identification codes and passwords. In this case, you need only type CAT and press the RETURN key. CAT, by the way, is the name of the modems Novation manufactures. CAT tells the system you want information on other bulletin boards. You will be given instructions on the screen that help you get the type of information you want.

Another source of information about local bulletin boards for Atari computer owners are magazines like *Antic* and *Analog*. Most Atari-oriented magazines carry articles about telecommunications and publish information on local bulletin boards that cater to the Atari computer owner.

A third source of information is *The Computer Shopper*. This monthly magazine is published in a newspaper format, and each issue has a comprehensive listing of active computer clubs and local bulletin boards. The address of many Atari clubs and the phone numbers of quite a few Atari oriented bulletin boards are listed in each issue of *The Computer Shopper*. That publication and the other magazines mentioned above are available at many newsstands and bookstores.

The SYSOPS of most bulletin boards are dedicated, hard-working people who invest time, effort, and often money into their bulletin boards so other people can use them. We all owe a debt of thanks and appreciation to SYSOPS who run Atari-oriented bulletin boards and Atari Special Interest Groups on the information utilities.

If the idea of running a bulletin board appeals to you, a program called *FoREM ST* (Commnet Systems, $99,95) will turn your Atari ST into a bulletin board system people can call. FoREM ST will work on a 520 or 1040 ST with two floppy disk drives, but most successful CBBS use a hard disk drive. The program will work with most modems that have an auto-answer feature, but it is designed to work best with a modem that is Hayes compatible. FoREM ST supports most of the features you expect to find on a bulletin board system including electronic mail, file uploading and downloading, message and conferencing facilities, and databases with different types of information. Many of these features are discussed in the remaining sections of this chapter.

Electronic Mail

When the Pony Express established a fast link between St. Louis and California, people cheered because mail from the East arrived in California in a matter of weeks instead of months. Speed, however, is a relative term, particularly when we are dealing with the transmission of information.

Regular mail today is fast when compared to the Pony Express, but there are times when you want something delivered faster. Overnight services such as the Postal Service's *Express Mail* and Federal Express' *Overnight Letter* are now available. If that isn't fast enough, Federal Express would like you to try *Zap Mail*, which is guaranteed to arrive within two hours of transmission.

Services such as Zap Mail are expensive, however. You can easily pay $30 or more to send a three page report to a colleague in another state. Fortunately there is a much less expensive and more convenient method of getting letters, reports, and documents from one place to another. Several companies, including MCI and Western Union, offer electronic mail services that can be accessed through your ST.

Although they vary considerably in their method of operation, all electronic mail systems increase the speed of information flow by transferring electronic signals instead of paper. With Federal Express' Zap Mail, the original documents are placed in a facsimile machine that creates an electronic image of each page. That electronic image is transmitted by satellite to a downlink station near the destination. Electronic mail systems that transmit facsimiles of the original documents can transfer images, such as drawings or graphics, as well as ordinary text.

Other electronic mail services, such as Western Union's *EasyLink*, bypass

the facsimile machines. Instead, they require you to input the information to be transmitted via your computer keyboard. That data is then transmitted to a printer near the destination, and a printed copy is either hand delivered to the recipient--expensive but fast--or dropped in the regular mail for delivery the next day. Systems such as EasyLink generally transmit only regular text. They cannot deal with charts, diagrams, or graphics.

Still other systems create electronic mail boxes as part of a larger information service. If you subscribe to general purpose information utilities such as The Source or CompuServe, for example, you are notified each time you use the service if a message has been sent to your electronic mailbox. Any message sent to you can be displayed on the screen of your computer and output to your printer. You can also transmit messages to other users of the information utility if you know their "address" or "handle" since it also serves as the address of their electronic mailbox. Both MCI and the EasyLink service of Western Union also offer this type of electronic mail service as well as hand-delivered options.

As you can see, the term "electronic mail" describes several different types of services. Some large organizations even have corporate electronic mail systems that allow you to transmit messages from the computer or terminal on your desk to the unit on a colleague's desk. In some systems, this service is restricted to the building or plant site where you are located. Others allow you to send electronic mail to offices in other states or countries.

Many local bulletin boards for Atari owners also have electronic mail services for callers, and the Atari computer Special Interest Groups (SIGs) on national information utilities like The Source, CompuServe, and GEnie also have electronic mail and "messaging" services for members.

Electronic Networking

In the jargon of corporate and personal computing, the term network has several meanings. If you network several computers in your office or classroom, each computer can communicate with the devices on the network such as printers, terminals, disk drives, and other computers. The term local area network, or LAN, is generally used to describe the hardware/software packages that let you connect two or more computers together so they can communicate with each other and share peripherals such as printers and disk drives. ST computers are particularly easy to network, and by the time you read this, several companies should have ST-compatible networking accessories that let you connect several computers to a common printer and set of disk drives.

One company, Quantum Microsystems, has already announced *ST-NET*, a local area network system for the Atari ST. It uses the MIDI port on the ST to connect two or more ST's to other computers as well as printers, hard disk drives, and other peripherals. The software for ST-NET is $149.95, and each computer must have an interface that costs $49.95.

The term network is also applied to the long distance telephone and microwave systems that allow computers and terminals located all over the country, and the world, to communicate with each other. You may use the services of communications networks such as *Telenet*, *Tymnet*, or *Uninet* when you send and receive data from information utilities such as CompuServe in Ohio or The Source in Virginia.

Although the two types of networks described above are important to many personal computer owners, they don't deal with the most important type. The networks we are referring to are networks of humans, not machines. The concept of

networking as a human process involving sharing resources, experiences, information, and support is not unique to the personal computing clan.

In fact, networking was a developing phenomenon well before personal computing became a cultural event. Howard Jarvis created a network of concerned California citizens who changed the state's tax laws in spite of opposition from most of the established power groups in the state. Regardless of whether you agree with Jarvis' views on tax reform, you have to admire his ability to start with a small group of people sitting around a kitchen table and build a network of thousands of volunteers who were all working for the same goal.

Computers, when used with telecommunications software, allow people living thousands of miles apart to meet electronically to discuss issues, share ideas, exchange information, develop plans, and coordinate projects.

For many groups a community bulletin board system has become an electronic meeting place. Others use one of the electronic mail systems as a conduit for exchanging information. There are, however, several types of telecommunications services that seem particularly suited to networking.

PARTI. One information utility, The Source, offers a service called PARTI, which is short for Participate. PARTI is an ongoing electronic forum for discussing a variety of topics such as world change, software piracy, and nuclear power. Any Source subscriber can join a discussion group (or create a new one) and read the comments made by others as well as add comments of their own.

Special Interest Groups. The Source, CompuServe, GEnie, and several other information utilities make it easy for individuals with special interests to communicate with each other by computer. CompuServe, The Source, and GEnie all have special interest groups (SIGs) for owners of many different types of computers, for writers, educators, investors, and many other groups. The Atari interest groups on these services are an outstanding source of information. The SIGs have information about new software and peripherals, notices of problems or bugs in software, product reviews, discussions on topics such as the use of CP/M on the ST, and tips on how to deal with common problems. Most SIGs also have large libraries of software you can download into your computer and use.

Private Sector and EIES. The Source also has a service called Private Sector for organizations that want to create a private teleconferencing service. A similar service is also offered by Electronic Information Exchange System (EIES). Both allow you to hold meetings on line with members or officers of your group without the expense of having everyone travel to one location. Each participant dials a telephone number and types the correct passwords and commands on their keyboard. Then they can read the comments made by other participants and make their own contributions by typing on their own keyboard.

CB. Lest you think networking is useful only when you have a cause to pursue, let us tell you about CB on CompuServe. CompuServe, a major general purpose information utility, has one service that is roughly the computerized equivalent of citizen's band radio. CompuServe maintains over thirty "channels" where people can meet and talk about anything they like. If you enjoy talking to other personal computer enthusiasts, CompuServe's CB service may interest you.

Electronic Publishing

Many small groups publish newsletters and bulletins for their members. The cost of preparing, printing, and mailing newsletters is, in fact, a major drain on the treasury of many organizations.

Fortunately, there is an electronic method of getting special interest publi-

cations to readers. Both CompuServe and The Source allow individuals and groups to "publish" newsletters and bulletins by storing the publications in the memory of the service's computer. You then have access to the material anytime you dial up the information service.

Although the two general purpose information services, The Source and CompuServe, offer some of the most convenient opportunities for electronic publishing, there are many others as well. One service, NewsNet, carries the full text of over a hundred specialized newsletters. You can get the latest information on who is bidding for transponders on the newest communications satellite or find out which oil company just made a big strike by reading some of the newsletters carried on NewsNet.

Consumer Services

Need to check airline schedules or make hotel, car rental, or resort reservations? You can do all that and more on many services. The Source, for example, has a want-ad bulletin board where you can shop for bargains (or advertise a bargain yourself) in everything from antique British cars to rental apartments in Sausalito.

Comp-U-Store. One service, Comp-U-Store, is a discount buying service that lets you select brand name products from the service and pay for them with your credit card. Comp-U-Store usually has excellent prices on software and accessories for the ST. There are also specialized buying services for products such as books, records, and tapes. Thus far, we are not overly impressed by most of these buying services. Some do offer attractive prices, but if you live in a city of any size, careful shopping locally can usually turn up the same product at a competitive price.

Banking Services. A great deal of press coverage has also been devoted to computerized banking services, but they are still somewhat experimental. Large banks like the Bank of America, Citibank, Chase Manhattan, and many others now offer banking by computer. Most services allow you to transfer funds from one account to another and to pay bills by computer.

That is a great idea, but there are problems. For example, with the Bank of America system, you can only pay bills to businesses on a list supplied by the bank. Few of the companies we do business with were on the list, a factor that makes computerized banking more of a novelty than a useful service for us. There are indications, however, that 1987 may be a year of change for computer banking. By the time you read this, the services will probably have expanded and may be well worth investigating.

Financial Services

Banking by computer is still in its developmental phase, but things are in high gear when it comes to many other financial services. You can check the current status of stocks, bonds, and commodities on several information utilities. In fact, you can create your own investment portfolio on many and the service will keep track of its performance.

A few information services, including the Dow Jones News/Retrieval Service are aimed specifically at people who need quick access to all sorts of financial and business information. Several brokerage firms now allow their customers to buy and sell stocks, bonds, and futures contracts via computer. Charles Schwab, one of the most successful discount brokerage houses, was also one of the first to

offer computerized brokerage services, but many other brokerage houses have joined Schwab in the computerized market.

Downloading Programs

A few services sell computer programs by phone. If you want to buy one of these programs, the service will transmit it by phone to your computer and charge the cost of the program to your account. Once it is in the memory of your computer, you can transfer a copy of the program to a disk.

Although "programs by phone" has been discussed and written about for several years, there are very few such services in operation today. The process requires sophisticated telecommunications software, and there is always the possibility a program transferred by phone won't work with your equipment. Most services mail a backup copy of the program and a manual, but the commercial software-by-phone ventures are still not very popular.

Software by phone is, however, a major means of distributing public domain and user supported software. (Software is in the public domain if the person who wrote it decides to give it away rather than copyright the program and sell it for profit.) Many bulletin board systems have thousands of public domain programs available for downloading. There is probably a bulletin board system in your area with programs that run on the ST.

Public domain programs are, by definition, free of charge--although some distribution services charge a handling fee. Another group of programs, sometimes called freeware, shareware, or user-supported software, are also low or no cost. These programs are the property of the distributing company, but they are available at low or no cost. The distributor usually asks you to pay a minimal fee, such as $10, for the program. If you find the program useful, you are asked to voluntarily pay a higher price, often around $30, for the software.

Many user-supported programs are available from bulletin board systems for downloading. This "honor" system of distributing programs makes the software available to a wide range of computer users without the massive expense of national advertising campaigns and multi-tiered distribution networks.

Some of the programs you download from bulletin boards or information utilities may also be "squeezed." A squeezed program has been compressed using a special coding technique so it takes up less room in the computer's memory and takes less time to download. Before using a file that has been squeezed, you will need to unsqueeze it with a special program called Squeeze or Unsqueeze. Some systems use the term archive instead of squeeze. An archived file often has the extension ARC in the file name. You must unarchive the file before it can be used. That is accomplished with a program called Archive that will usually be available on any bulletin board that archives files. Archive itself is not compressed, so you can download it and use it to unarchive other files.

Computing

It may seem redundant to say a computer can be used for computing, but this is a little different. Your ST is quite capable of performing most, if not all, the computing tasks you or your business require. There may be times, however, when you need more computing power than is available in your personal unit. Most of us can't afford a two million dollar IBM mainframe for occasional use, but an ST in our home or office can also be connected, by phone, to a large mainframe computer. That connection lets you use the computing power of the larger com-

puter for a nominal charge per minute. Many local computer services sell "timesharing" on their systems, and a growing number of universities let students harness the power of the school's mainframe system from the keyboard of a personal computer. Students at many schools can, for example, write and run programs written in a variety of languages including BASIC, COBOL, Pascal, and FORTRAN.

Many of the national information services also make available quite a few canned programs of their own. The programs available include games, business software such as loan amortization programs, and software for special applications such as statistical analysis of business or scientific data. You cannot buy these programs and run them without being connected to the information service. In essence you rent them by connecting to the service by phone and typing in the name of the program you want to use.

TELECOMMUNICATION SERVICES FOR ST OWNERS

As discussed earlier, there are a number of bulletin boards and special interest groups that cater to owners of Atari computers. Most of these services even have special sections for ST owners. Information utilities with a number of Atari-specific services are discussed in this section.

CompuServe. There are three general purpose information utilities today that offer services in all the categories mentioned above. The largest is CompuServe, which is owned by H & R Block. Sign up kits for CompuServe are available in many computer stores and book stores or you can write CompuServe. You can also call CompuServe and charge the cost of a sign up kit (about $30) to your credit card. CompuServe has several Atari-oriented special interest groups. There is one for owners of the 8-bit Atari models, another for ST owners, and groups for professional programmers developing software for the ST. Antic magazine also sponsors several interest groups on CompuServe and has a special section on the ST.

The Source. CompuServe's major competitor is The Source, which is owned by Reader's Digest. These two "information utilities" offer somewhat similar services and generally charge between $7 and $12 for each hour of connect time. CompuServe's special interest groups for Atari owners are some of the largest and most active, but The Source also has Atari oriented interest groups.

GEnie. In 1985 another general purpose information utility was launched by General Electric. GEnie was just getting off the ground when this chapter was written, but the utility is likely to attract many subscribers because it charges only $5 an hour for access in the evening on weekends. Most people access information utilities with modems that work at 300 bits per second (BPS) or 1200 BPS. Many utilities charge extra if you transmit and receive data at the faster 1200 BPS rate. GEnie does not. The low price of accessing GEnie is one reason to check out the service. For Atari ST owners, another is the special interest group on GEnie for Atari computer owners.

Delphi. Although the Delphi information utility has been in operation for several years, it has never enjoyed the high visibility and popularity of services such as CompuServe. One reason is that Delphi has never been heavily advertised. Word of mouth and articles on Delphi in computer magazines have always been the most likely way someone would hear about Delphi. There are several reasons ST owners should consider Delphi. It offers many of the same services available on The Source and CompuServe but charges less for each hour of use. In addition, *Analog*

magazine, one of the better Atari-oriented publications, supports an Atari Users' Group. That service has an electronic forum where Atari computer owners can swap information and ideas, a database of public domain programs you can download and use, and a conferencing feature that has on-going discussions of many different topics.

AN ST SAMPLER

It is hard to imagine how much information and software for the ST is available today in the memory banks of information utilities such as CompuServe and Delphi. A small sampling of the electronic smorgasborg you will find on Compuserve is presented in this section. The material was exerpted from SIGATARI, the Atari Special Interest Group on CompuServe. SIGATARI has four different sections:

1 Atari 8-Bit Forum
2 Atari 16-Bit Forum
3 Atari Developers Forum
4 ANTIC ONLINE (W)

The material below is part of the introduction to the forum for ST owners and a list of a few of the programs and data files available.

ATARI 16-bit Forum
March 1986

Welcome to the ATARI 16-bit Forum. It is one of 3 SIGs or (S)pecial (I)nterest (G)roups, for ATARI computer enthusiasts on CIS. This area has been established as a national forum specializing in programs and information pertaining to the ST-series (16-bit) of computers. This forum is not affiliated with ATARI Corp, and is maintained by independent SYSOPS under an agreement with the CompuServe Information Service.

Access to the information and use of these facilities is available to CIS customers requesting membership through the online signup option. By requesting membership, it is understood that you have read this information and agree to abide by the general policies listed below. There is no additional charge for membership or use of the Forum facilities beyond the applicable CompuServe connect charges.

To signup for membership in this SIG, select the appropriate menu item on the NEW MEMBER MENU. You will be prompted to enter your name. SIG POLICY REQUIRES USE OF YOUR FULL, REAL NAME. "HANDLES" OR FIRST NAMES ONLY ARE NOT SUFFICIENT FOR CONTINUED MEMBERSHIP. CORPORATE NAMES ARE PERMITTED, SUBJECT TO ACCEPTANCE BY THE PRIMARY SYSOP.

Membership privileges are enabled automatically, and remain in effect unless specifically deleted by the SYSOPS as long as you access the Forum at least once every 60 days. The membership file is periodically purged of inactive members to improve system operation. If this should occur, all you need to do is sign up again and your membership privileges will be quickly enabled.

Forum members may access the (CO)nferencing channels, DATA LIBRARIES, and message boards which have been broken down into sub-topics. To see a listing of the sub-topic names, type "SB" upon entry to the Forum.

Our DATA LIBRARIES contain many programs, textfiles, HELP info, product reviews, and transcripts of previous COnferences held in the Forum. These files are available for DOWNLOADING at no additional charge to SIG members whose terminal software supports one of the various file transfer methods, including XMODEM protocol.

Online HELP is available at virtually every level of the Forum. When in doubt, try entering a "?" or the word "HELP." All commands should be typed from the keyboard and terminated by a [RETURN].

Questions about your ATARI computer or peripherals or about the CompuServe system may be left in the appropriate message section addressed to a specific individual, to:ALL, or to the SYSOPS. The true value of SIG*ATARI as an invaluable source of information is directly attributable to the skills and knowledge of the membership. Many industry professionals and authors are active members of the SIG willing to lend a hand to others.

The SYSOPs of SIG*ATARI were chosen for their broad backgrounds with ATARI computers and the CompuServe system. Most are volunteers, who do this in addition to their regular occupations. They stand ready to help you with your ATARI or Compuserve, and if they can't give you the answer, they'll try to point you in the right direction.

SYSOP*Ron Luks 76703,254
SYSOP*Steve Ahlstrom 76703,2006
SYSOP*Russ Wetmore 76703,2010
SYSOP*Richard Brudzynski 76703,2011
SYSOP*Dave Groves 76703,4223
SYSOP*Tom Hudson 76703,4224

When you reach the 16-bit forum, you have access to many different types of information. There are articles on many different topics, programs you can download, files of data for generating many different types of music and art. We have included a small sampling below of the type of information you will find in ATARISIG. All the entries that follow were taken from the Atari 16-Bit Forum which deals specifically with the ST. Each entry begins with the CompuServe number of the person who uploaded the file. For example, an entry that begins with [76566,1425] is one that was uploaded by one of the authors of this book since that is his CompuServe number. The next line in each entry contains three pieces of information: the name of the file, the date it was uploaded, and the number of times it has been accessed.

The next line in the listing is the keywords for this file. CompuServe has a search feature that lets you type a keyword and ask it to find all the listings that relate to that keyword. The listings below also include a brief description of the file. They are obtained by selecting the Browse option on CompuServe. If the description indicates the file is one you would like to download, you can set your telecommunications program up for downloading and then tell CompuServe which file you want to download by typing the name of the file.

[72355,407]
WEFAX.TXT 02-Jul-86 6095 Accesses: 32

Keywords: WEFAX WEATHER FACSIMILE ANTIC SATELLITE PHOTOGRAPHY

An announcement concerning WEFAX, a weather-facsimile program for 8- and 16-bit Atari computers to be published in the September, 1986 Antic magazine. This program enables Atari computers, when interfaced to a shortwave receiver, to receive weather satellite photos. Also included in this text file is a review of this software/hardware package, and a comparison to the version for the Tandy Color Computer (CoCo) that has been available for the past year or so.

[73547,3520]
TDIBUG.TXT 29-Jun-86 15820 Accesses: 12

Keywords: TDI MODULA-2 TYPOS SUGGESTIONS QUESTIONS FAULTS

This file contains a list of the typos, textual faults, system faults, questions and suggestions I have come up with after 3 weeks with v2.00A of the developer's Modula-2 from TDI. Mostly minor stuff.

[72237,1015]
MIDI.BBS 01-Jun-86 880 Accesses: 51

Keywords: MIDI BBS COMPUTER MUSIC

Invitation to join American MIDI Users Group (formerly Dallas MIDI Users Group). Discuss Musical Instrument Digital Interface, swap Synth patches, etc with members all over the U.S.

[76703,4224]
WAACE.PI3 19-Jun-86 81665(34848) Accesses: 19

Keywords: CAD-3D WAACE USER GROUP IMAGE

This is a kind of user group logo I did up -- it's for the Washington Area Atari Computer Enthusiasts (WAACE), and shows the WAACE lettering alongside the Washington Monument. View with DEGAS or the SHOPIC program. Created by Tom Hudson.

[73137,773]
KLINGO.NEO 15-Jun-86 81985(35008) Accesses: 68

Keywords: KLINGON

THIS IS A NEOCHROME PICTURE OF A KLINGON BATTLECRUISER OVER A PLANET. DONE BY WILLIAM HEIM.

[76703,4224]
TABLE.PI1 15-Jun-86 81665(34848) Accesses: 61

Keywords: DEGAS LOW-RES TABLE CAD-3D

This is a scene of a table and two chairs, created with CAD-3D (TABLE.3D in DL5 is the 3D file). On the table is a vase with three cattails, and a sloppy youngster has left his bat and baseball on the table. Created using CAD-3d, then touched-up with DEGAS (anti-aliasing). Enjoy! Created by Tom Hudson

[72315,1522]
SCROOG.BAS 05-Jul-86 5180 Accesses: 23

Keywords: MONTHLY FINANCES ST BASIC

SCROOGE: A simple monthly bill-paying manager; improvements and elaborations to come! Runs in ST BASIC. Your comments and suggestionswelcome (72315,1522).

[74516,3531]
SDFFMT.DOC 04-Jul-86 7580 Accesses: 13

Keywords: DBASEHDBASE CONVERT

This is a combo file on how to modify and convert dbase 3 plus files and programs to run under H & D-base on the ST, Contains SDF conversion mini-program source. This will run as a Dbase 3 Plus program on IBM-PC to do the initial conversion before moving files to ST..

[73267,3560]
NAVIGA.PRG 29-Jun-86 161040(68736) Accesses: 27

Keywords: NAVIGATION FLIGHT PLANNING AVIATION

Navigate! is a GEM flight planning program written in Personal Pascal. Requires ICAO.DAT to work (ICAO.DAT must be on ram disk D:). Requires Sample.FP. See Navigate.DOC for instructions.

[76703,1077]
CHESS.3D 29-Jun-86 454820(194080) Accesses: 15

Keywords: CAD-3D CHESS SET OBJECT FILE 1040ST ONLY

A complete Chess Set! Tom Hudson's largest CAD-3D object yet! Size: 178K! (1399 blocks! 40 min. dl @ 1200b). Requires 1MGB RAM. Set to Edges Only before loading; for a good view, try: up 20, left 40. Select just a few objects (knight, bishop, rook) and look at the incredible detail in the more complicated pieces close-up! For a surprise, select just the white

king, and then turn a light on bottom and rotate to view the base.

[75156,3131]
FDI.TOS 20-May-86 141855(60544) Accesses: 227

Keywords: FLOPPY DISK INDEX CATALOG LIBRARY

This is a Floppy Disk Indexing program which builds and maintains a catalog of files on "floppy" disks. Download using XMODEM and rename to FDI.TOS. Read FDI.DOC elsewhere in this SIG before using.

[72457,1217]
PAN109.CMP 11-May-86 24735(10560) Accesses: 76

Keywords: PRINTER DRIVER 1ST WORD PANASONIC 1091 1092

1st Word (version 1.03 or later) printer configuration file for the Panasonic 1091/1092. Same as PAN109.HEX except when printing in NLQ super/subscript will be in compressed print (looks a little better). To use with earlier versions, load file in 1st Word and delete one of the 6 zeros under the Printer Configuration Table. Rename PAN109.HEX

[72747,3600]
CANOND.SNG 07-Jul-86 14640(6272) Accesses: 21

Keywords: MUSIC STUDIO CANON IN D JOHANN PACHELBEL

ANOTHER OF MY MUSIC STUDIO FILES. THE BEAUTIFUL "CANON IN D" BY JOHANN PACHELBEL. CONFIGURED FOR PIPE ORGAN ON MY ROLAND JUNO 2. PICK SOMETHINGSIMILAR, OR USE STRING ENSEMBLE. THOSE WITH MULTI-TIMBRAL MIDI'S CAN SPLIT THE VOICES ACROSS VARIOUS PRESETS FOR EVEN NICER SOUND. ENJOY! BRUCE SMITH

[75776,711]
CHARIO.SNG 04-Jul-86 11715(5024) Accesses: 50

Keywords: CHARIOTS OF FIRE

THEM SONG FROM THE MOVIE CHARIOTS OF FIRE. IT IS 36 BLOCKS LONG. IT WILL OVER WHELM A SINGLE KEYBOARD. IT WAS SET-UP FOR A CASIO CZ-230S ON THE BLACK NOTES A CASIO CZ 3000 ON THE RED NOTES AND A CZ-101 ON THE BLUE NOTES USING A ROLAND MIDI THRU BOS AND A SONY MUTLE AUDIO MIXER INTO A SONY ES SOUND SYSTEM IT WILL TARE DOWN THE WALLS IF YOUR SOUND SYSTEM CAN TAKE IT.
OF CORSE IF ONLY ONE KEYBOARD IS USED TURN OFF THE RED AND BLUE NOTES WITH THE TRACKS FETURE.

[75776,711]
GRANFL.SNG 28-Jun-86 14975(6400) Accesses: 44

Keywords: YOURE A GRAND OLD FLAG

SONG FROM THE MOVIE YANKEE DOODLE BOY BY GEORGE COHAN. IT WAS ENTERED INTO THE MUSIC STUDIO USING THE TRACKS FEATURES AND SOUNDS BEST USING A PIANO VOICE BASE INSTRUMENT AND A VOX OR DOUBLE REED FOR THE LEAD. AND IT SOUNDS BEST USING TWO OR MORE SYSTHESISERS . IF ONLY ONE IS USED THE SOLO VOICE CAN BE TURNED OFF BY KILLING THE RED NOTES. THE RESTS AT TOP ARE TO KEEP THE TIMING STRAIGHT .HAVE A HAPPY --<<4TH>>-- GANG

[70007,1200]
PHILDR.DAT 22-Jun-86 980(448) Accesses: 53

Keywords: CZ101 PATCH CASIO DRUMS MUSIC SYNTHESIZER

This is a drum-like patch that I did on a Casio CZ101 synthesizer. I called it PhilDrum because it sounds like the drums Phil Collins uses on the No Jacket Required album(especially in the middle of the keyboard-- the lower notes sound more like a bass drum). If anyone has any other interesting patches, such as a REAL PIANO or SAXOPHONE, please_ upload them. When I get more, I will too. Comments to 70007,1200. -------Bob.

[72605,1457]
HARP1.DAT 22-Jun-86 980(448) Accesses: 29

Keywords: CASIO PATCH - HARPSICHORD SOUND

This one sounds more like a cross between a synthesized piano and a harpsichord Use with the Casio Patch program found elsewhere in this SIG. Enjoy!

- Paul Wu

[76367,200]
TWOPAR.SNG 22-Jun-86 8785(3776) Accesses: 91

Keywords: MUSIC STUDIO CASIO CZ-101 BACH CLASSICAL

Switched-On Atari This is a little tune by Old Bach, the Two-Part Invention in F major. It runs in Poly mode and uses Preset#1-Brass Ens.1.
The Soft Organ patch also sounds good. For those of you without a MIDI synth, the internal speaker can be used. Wendy Carlos, eat your heart out!
Walt Sullivan

[75776,711]
TOMORR.SNG 17-Jun-86 14975(6400) Accesses: 54

Keywords: TOMORROW FROM ANNIE

THE TOMORROW SONG FROM THE BROADWAY MUSICAL. IT IS WRITTEN FOR THE CASIO CZ 3000 USING ACCORDIAN VOICE BUT THE VIOLIN VOICE ALSO SOUNDS GREATE SO IF YOU USE ANOTHER SYNTHESIZER YOU CAN SET IT USING THE MIDI PRAMITERS BLOCK IN THE TOP LEFT OF THE MAIN MENU .

[75776,711]
WEWORL.SNG 14-Jun-86 26035(11136) Accesses: 76

Keywords: WE ARE THE WORLD SONG FOR MUSIC STUDIO

A SONG WRITTENEN BY MICHAEL JACKSON AND LIONEL RICHE FOR THE USA AFRICA CHARITY DRIVE. IT WAS WHITTEN FOR THE CASIO CZ 3000 USING THE HUMIN VOICE TWO GUITARS A PAINO AND A FAT BASE. YOU WILL HAVE TO RESET THE VOICES WITH THE MIDI PRAMITER MENUE TO MAKE IT SOUND GOOD ON ANOTHER SYNTHIZER OTHER THEN THE CZ-3000 AND THE CZ-101 AND THE 3000 ARE NOT THE SAME NUMBERS FOR THE SAME VOICES AND OF CORSE THE YAMAHA IS STILL DIFERENT

[72507,257]
MOZART.SNG 08-Jun-86 15620(6688) Accesses: 132

Keywords: MUSIC STUDIO WOLFGANG AMADEUS MOZART SONATA IN C-G

No, this is not "Rock me, Amadeus" by Falco but it is the melodius theme of the Allegro movement of the Sonata in C-G by Mozart. Use a piano sound if you have a MIDI synth but it also sounds nice with just the ST sound chip and the provided "piano rev"setting. Enjoy !

Ed Bartholomey
Home Computers Company
Towson, Maryland

[71777,2600]
ENTER.SNG 05-Jun-86 5215(2240) Accesses: 135

Keywords: MUSIC STUDIO SNG SONG THE ENTERTAINER BY SCOTT JOPLIN

"The Entertainer" by Scott Joplin, for the Music Studio by Activision. Uses the ST's internal voices. 16 blocks.

Kevin Engler
71777,2600

[70705,1000]
CASIO.PRG 04-Jun-86 77440(33056) Accesses: 50

Keywords: CASIO

This is a MIDI usable program for your Atari ST which allows you to upload and download information from your Casio synthesizer!

8

Home, Recreational, and Educational Applications

Not too long ago, Atari computers were often mistakenly considered expensive video arcade game machines. Admittedly there were a lot of those types of games available for Atari's XL and XE computer series but that was by no means the only application available. The sophistication of the ST should put the "video game machine" rumor to rest once and for all. Yet the ST does make a good home computer and there are already a number of good game programs available that will run on it. Those will be discussed later in this chapter. First, let's look at some of the other ways you might use your ST at home.

HOME SOFTWARE FOR THE ST

In an earlier chapter we discussed using the computer to help with all the records a business has to keep. Most individuals do not have the same number of financial and record-keeping responsibilities as a business, but today most of us must keep some financial and family records.

The most popular piece of equipment that helps families keep track of financial records is probably a very full shoebox. Between January and April 15 each year, millions of people go through the familiar routine of looking high and low for lost receipts, boxes of cancelled checks, and medical or insurance records which they are sure were in the bureau drawer "just last week." On April 16 each year, those same people vow tax time next year will be different. They will keep good records and organize them so that any piece of information can be found quickly when it comes time to complete their 1040 form. Some computer programs can help bring a little order to your records.

Keeping organized tax records is not the only record-keeping task you can assign to the ST. This section will discuss some of the general applications areas and some of the programs available. You may also want to read the chapter on business and professional software and the chapter on word processing since many of the programs described in those chapters are also used in the home. Most of the programs in this chapter were written for specific purposes. Two types of software discussed in the business chapter--database programs and electronic spreadsheets--can be used for some of the same kinds of tasks as these specific purpose programs but can be customized to meet individual needs.

Bank Account Management

For those who have a checking account, balancing the checkbook is sometimes classified as monthly torture. Some approach their cancelled checks, check stubs, and the printed statement from the bank scientifically, confident that each penny will be found. Others find this monthly task rarely turns out satisfactorily.

Whether you are one who is willing to spend an hour searching for an error or one who thinks any figure within $50 is good enough, a computer with a check-balancing program can speed up the process. It can help you find errors, alert you to checks that were written but have not been cashed, and keep a running record of the amount of money you are spending in various categories.

Did you know the Escort had $456 of repairs in the last 5 months or that you spent $132 dollars on video tape rentals since January? How much did you contribute to the United Way at the office? Bank account management programs don't generate new information for you, but they will organize what you have so it is easier to find and use.

The Check Minder Program (Haba, $74.95) displays a realistic check form. You fill in the information down to what the check is for and if the amount is tax deductible.

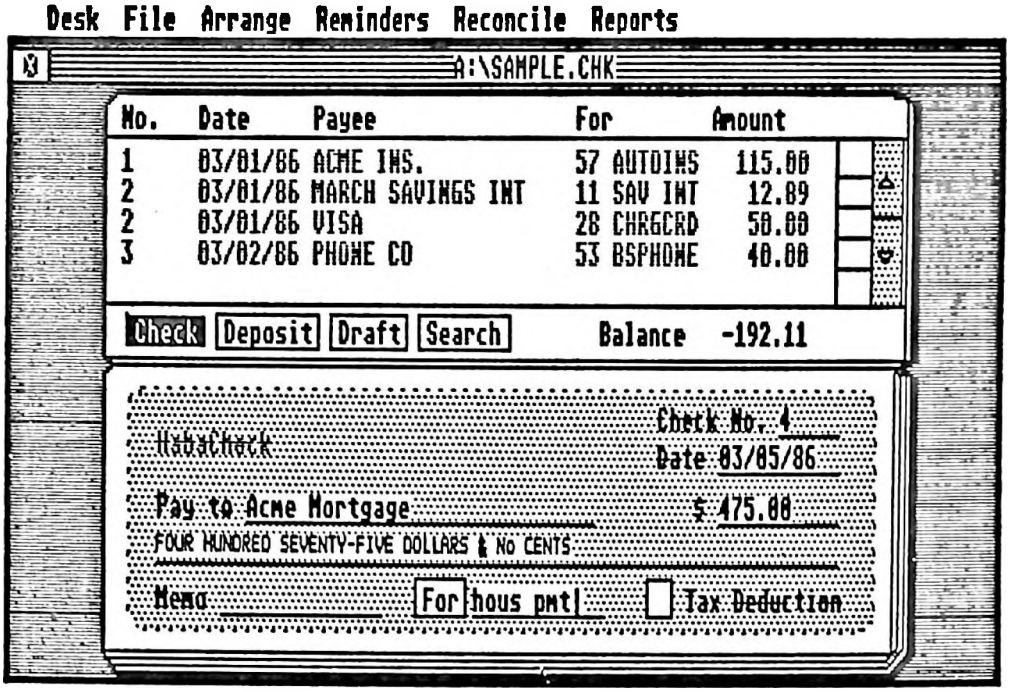

Desk File Arrange Reminders Reconcile Reports
A:\SAMPLE.CHK
No. Date Payee For Amount
1 03/01/86 ACME INS. 57 AUTOINS 115.00
2 03/01/86 MARCH SAVINGS INT 11 SAV INT 12.89
2 03/01/86 VISA 28 CHRGCRD 50.00
3 03/02/86 PHONE CO 53 BSPHONE 40.00
Check Deposit Draft Search Balance -192.11
HabaCheck
Check No. 4
Date 03/05/86
Pay to Acme Mortgage
$ 475.00
FOUR HUNDRED SEVENTY-FIVE DOLLARS & NO CENTS
Memo
For hous pmt
Tax Deduction

Figure 8.1–Screen Display of the check in the Check Minder program.

The program allows for a maximum of 16 Income categories and 80 Expense categories. The Check Minder disk also includes a sample file of standard IRS categories which make it easier to keep track of your income and expenses for tax purposes. You can use these categories and add others or make up your own list of categories.

Balancing your checkbook is easy with Check Minder. It will even tell you how much the total is off if the account is not in balance. If your house payment is due on the fifteenth can set up a reminder to help you remember to pay it on time. In fact, you can create a list of reminders with due date, who the check should be written to, the amount of the check and the Expense category it goes in. This pro-

gram will actually print your checks for you and display or print a variety of reports such as Income by Category, Income and Expense Statement, Average Cash Flow, or Tax Preparation List.

With *Personal Money Manager* (Michtron, $49.95), you can keep all your records for one whole year. You set up all your budget accounts, such as Car Expenses, Income, or Entertainment, with the Account Maintenance program. If you wish, you can divide your paycheck up into the accounts automatically. You can enter income and expenses into any account every day; or, if you prefer, you can wait and enter all the transactions for one week, or month, at the same time.

If you can't remember whether the car payment has been made or not, you can choose the Ledger Query option on the Process menu and check to see if that transactions is listed on the Car Expenses account. Of course if you mailed the check but didn't enter the transaction, this program won't help much. If you need to examine your records more closely or take them to the accountant, Money Manager will let you print a copy of your budget, monthly expense and income records, a list of the accounts with their balances and the amounts you budgeted, a list of monthly transactions by account, or a monthly report for only one account.

Keeping On Schedule

MichTron's *Calendar* (39.95) program documentation tells you how to transfer the program from the Calendar disk to your TOS MASTER disk. Then, you can activate the Calendar from the Desk menu any time since it is loaded whenever you use the GEM Desktop. This program will display a calendar for a full year, one month or one day. You can display any year from 1980 to 2099. Did you miss your last doctor's appointment? When is Sue's birthday?

If you want to set up reminders about these or other important dates, Calendar allows you to enter messages for each day. Reminders for regular weekly meetings can be entered once and automatically recorded for each week. The messages can be entered as general messages or as appointments in time slots from 12:00 A.M. to 11:45 P.M. If you enter times, you can also set an alarm to remind you--if you are working at your computer--when you need to quit and go to your appointment. When you display the calendar for a specific day, the messages appear. For reminders when you aren't at your computer, Calendar lets you print the display so you can put it under the magnet on the refrigerator door or in your attache case where it's handy.

Making Your Own Will

Making sure your possessions are taken care of if something happens to you is a major consideration. Several books are available which tell you how to write you own will, but *Solutions: Wills* (Haba Software, $49.95) is a software program which includes legal will forms which you simply fill out and print. There is a guide book which helps you make decisions about which form would best fit your circumstances.

RECREATIONAL SOFTWARE FOR THE ST

Several types of computer games and recreational software are discussed in this section. The first category, board games, consists of computerized versions of popular games such as backgammon. The second category, games of chance, includes card games like poker and blackjack.

Simulations, the third category, involve a real or make-believe environment. The environment may be a simulation of something real, such as a game of golf, or a fantasy environment with its own set of rules and regulations. We have divided simulations for the ST into two categories--adventure simulations and arcade simulations. Adventure simulations call for careful analysis of the situation and thoughtful decisions. Arcade simulations usually require good eye-hand coordination, sturdy wrists, a firm grip on the joystick, and good reflexes.

Board Games

Computerized board games take advantage of the excellent graphics features of the ST to display a picture of the board. In some cases, the computer takes the role of one player and you play against it.

Pegammon (Artworx Software, $17.95). This program is one of Artworx PX Software series of low-cost, original programs designed to sell for prices as low as $9.95. The Pegammon program is a backgammon game with a three dimensional board. Options allow for varying levels of game difficulty.

HippoBackgammon (Hippopotamus Software, $39.95). Backgammon against programmable robots is the game in this program. You can play against novice, intermediate or expert robots or have the robots play each other. The program is based on artificial intelligence which makes the robots appear to think and make decisions about their moves. However, their decisions and strategy can be changed to vary the game.

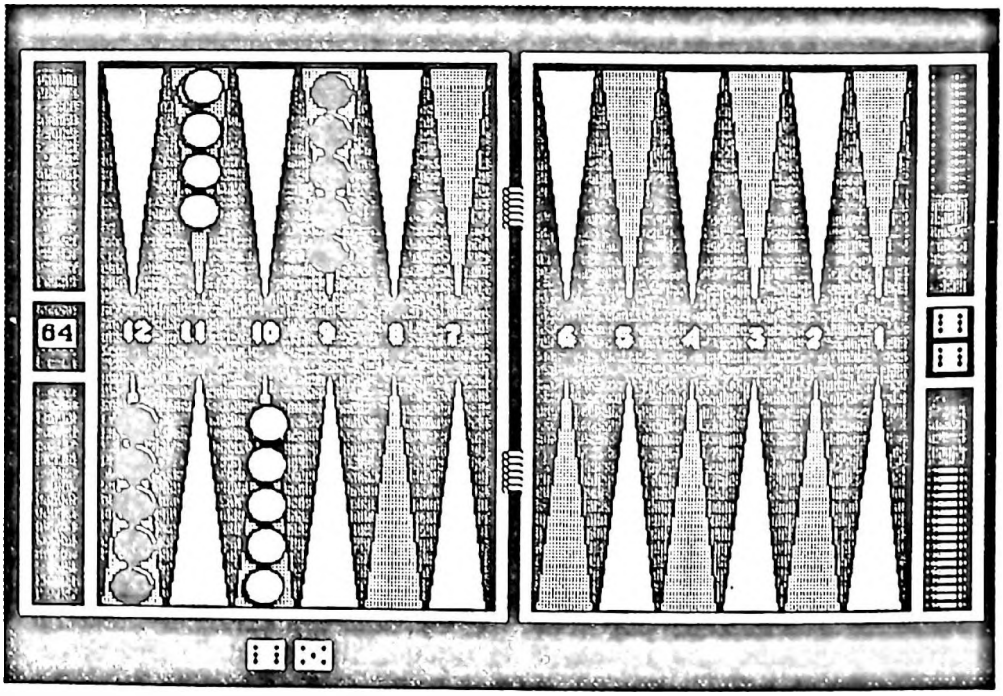


Figure 8.2-The HippoBackgammon Game Board.

Word for Word (Bay View Software, $39.95). This program is similar to the popular board game, Scrabble. Each player--up to four can play each game--must make words from the letters "drawn" on a crossword-type grid. New words must connect to words already played on the board. Each letter played has a point value and some squares on the board offer a bonus of doubling or tripling the word score. Word for Word is another program which offers three different skill lev-

els: beginner, intermediate, and advanced. It also allows you to play against the computer, with other players, or both. One option of the program even gives you the opportunity to create your own game board. Another lets you set a time limit on the game.

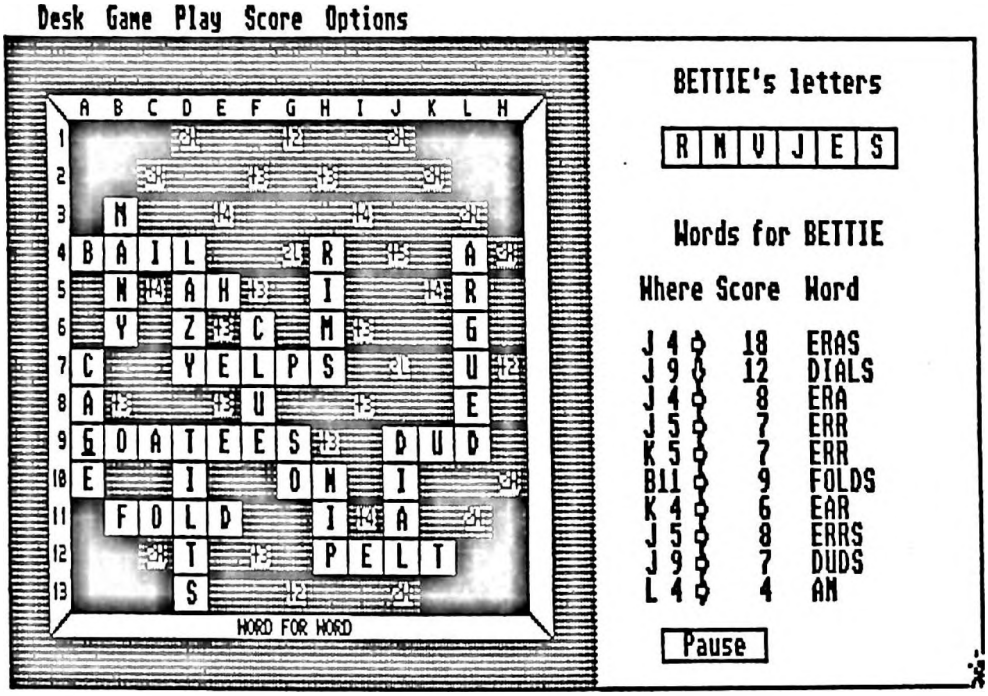


Figure 8.3-A Word for Word game between computer opponents Alphie and Bettie.

Games of Chance

Programs that computerize games such as poker, roulette and craps do not completely duplicate the operation of the electronic slot and poker machines in Las Vegas casinos, but you may be surprised by the quality of the display and the sophistication of the programs.

Strip Poker (Artworx Software, $39.95). This program does just what you think it does. You play against either Suzi or Melissa, two very attractive opponents who play by the rules and remove a piece of clothing each time they lose. The color graphics in this program are very detailed. If this were a movie, it would be R rated.

Additional opponents are available on separate disks for versions of Strip Poker used on other computers. They will probably be available for the ST by the time you read this.

Bridge 4.0 ($29.95). Another Artworx program, Bridge 4.0, displays cards as though they were on a bridge table. Bids and card selections are entered entirely with the mouse. The user plays the game of contract bridge with a computer partner and against two computer opponents. Any of the four players, the user or the computer players, can take the bid and play out the hand. The program keeps track of the score and bidding and will tell you if you make an illegal bid or play.

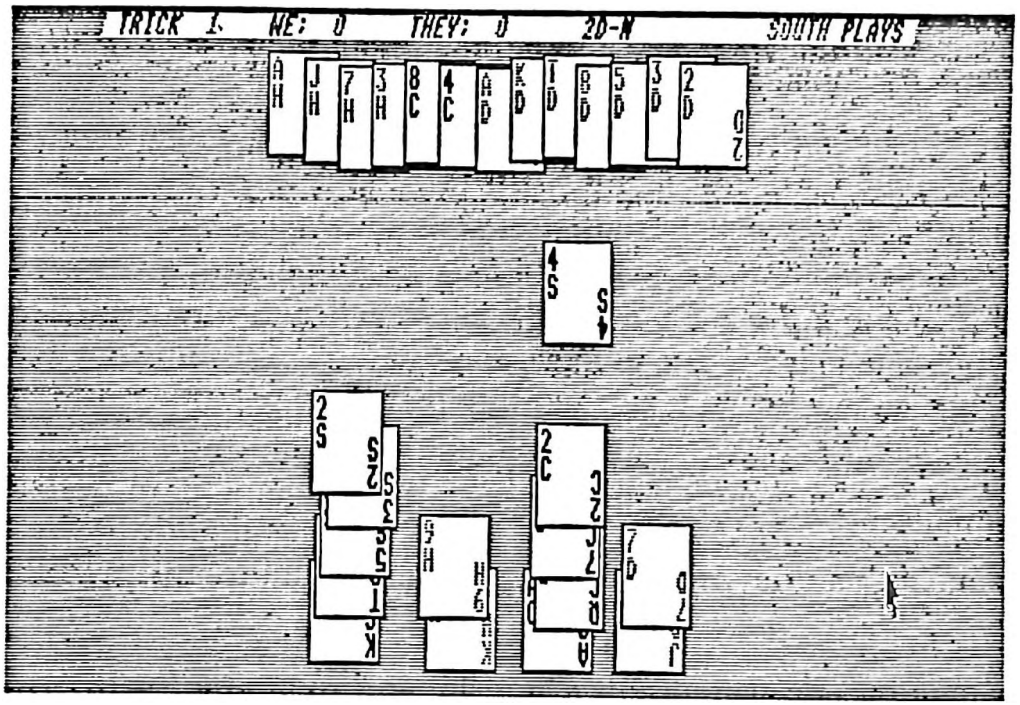


Figure 8.4-South leads the four of spades in this Bridge 4.0 hand.

Compubridge ($29.95). Artworx designed Compubridge to teach the beginner to play bridge. However, it is also a good way for the more experienced player to

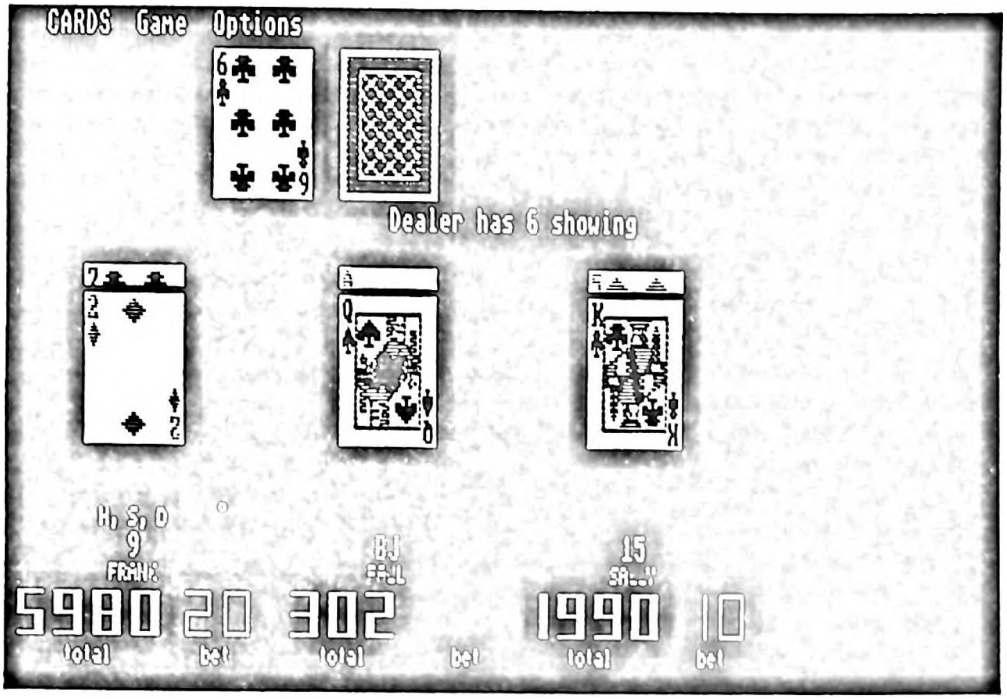


Figure 8.5-Three players at the blackjack table in MichTron's Cards.

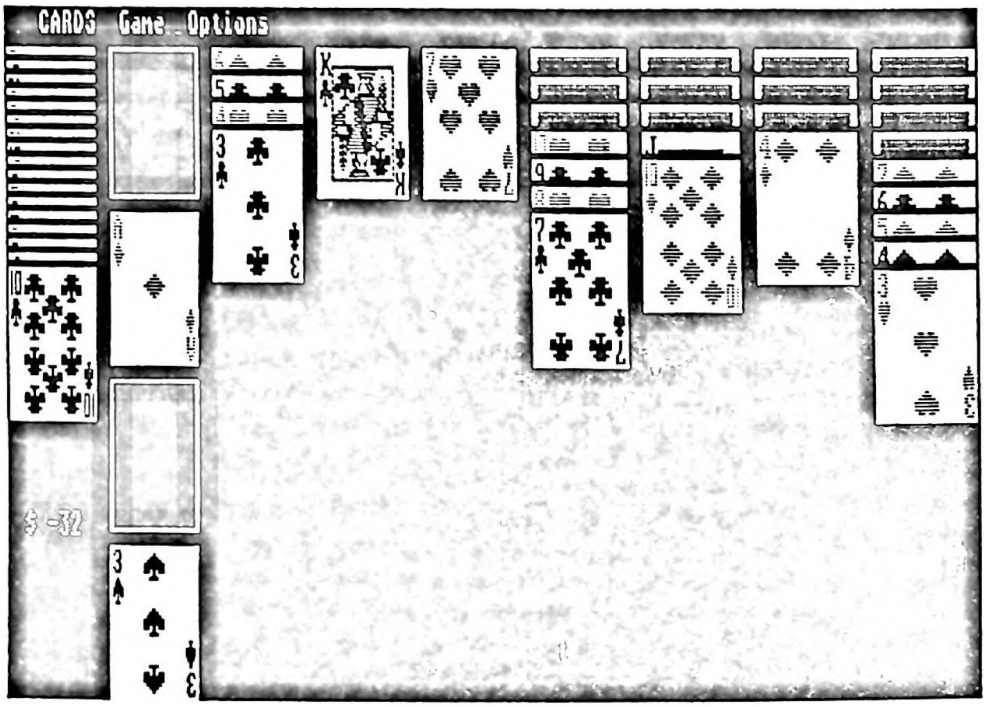


Figure 8.6-The Klondike Solitaire Game from MichTron's Cards.

practice and build skills. The program is designed around ten chapters of text--based on a bridge teacher's manual--and eight quizzes which consist of random skill-building problems. Compubridge will help sharpen your skills by correcting mistakes or giving advice about strategies.

Cards (MichTron, $39.95). This software package has five different card games: Blackjack, Cribbage, Klondike, Poker Squares, and Solitaire. Blackjack works like the Las Vegas casino game. Each player sets the amount of money he has and how much he wants to bet on each hand. Players bet against the dealer and win when the total of their cards is closest to 21. Cribbage is also designed for more than one player to play against the computer. Whoever is first to reach 121 in this computer version of the classic card game wins. Klondike is a one-player game involving playing the cards onto "building piles" in reverse order--from King down to Ace--in alternating colors. Poker Squares displays a five-card by five-card grid and the player builds poker hands by inserting the cards--dealt one at a time--into the grid. Scores depend on the hands which result in the vertical and horizontal rows of the grid when all the spaces in the grid are filled. Solitaire consists of playing cards in ascending numerical order by suits much like the old familiar solitaire game. The cards are dealt in groups of threes with only the top card available for play. Cards not played on the first round can be dealt two more times then the game ends if all the cards are not played onto the suit piles.

Electro Solitaire and 21 (Softlogik Corp., $19.95). Here is another package deal containing the Klondike solitaire game and casino-style Blackjack.

Adventure Simulations

Most arcade video games have detailed graphics and use joysticks or game paddles. They also rely on excellent eye/hand coordination, fast reflexes, and good peripheral vision for success. Adventure simulations, on the other hand, are strat-

egy oriented and require you to think your way through rather than act your way through. Many of the adventure simulations do not even have graphics displays. They rely primarily, or entirely, on text displayed on the screen.

All adventure simulations put you, the character, in an environment--forest, cave with many chambers, castle with many rooms, maze. Then they give you a task to perform--stay alive, get out, find the treasure, rescue the prisoners--and put all sorts of dangers in your way--monsters, cliffs, sorcerers, enemy soldiers, bombs. Two final ingredients of an adventure simulation are incomplete information and clues to getting more information.

A game is likely to begin with a paragraph that sets the scene for the simulation such as the following: "You are standing on the edge of a forest. There is a narrow winding road that leads out of the forest and winds through the hills to a large stone house. The house is shrouded in fog and stands on the edge of a cliff. You see a bootprint!"

Most adventures require you to move about through a series of "rooms." The term room, however, really refers to different segments of the simulation such as rooms, cities, chambers in a cave, and so on. There are ways to learn more about the simulation's environment and about ways to successfully deal with the adventure. Adventure simulations let you explore and move about the rooms by giving the computer instructions--WALK NORTH, OPEN DOOR, LOOK AROUND, ENTER. Some adventures understand only a few instructions while others understand several hundred.

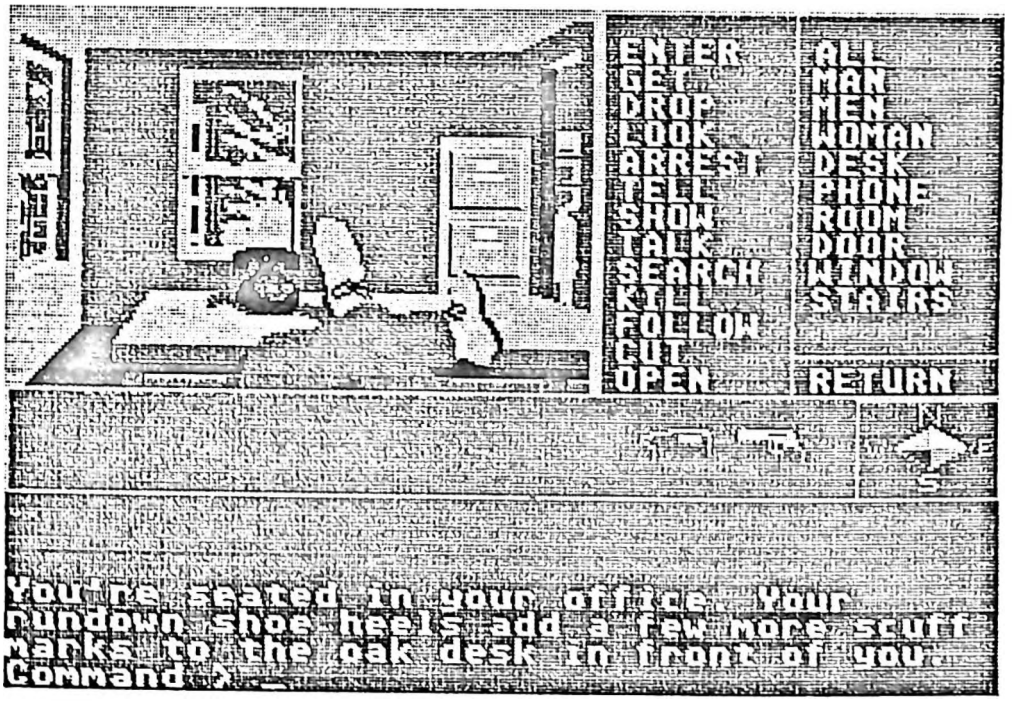


Figure 8.7-The opening scene from Activision's Borrowed Time.

You, and the other characters, can earn various powers and encounter foes not likely to be overcome. Things do not always work out as you hope, however, and it is not unusual to lose some of the characters you have developed along the way.

Some adventure simulations take advantage of the ST's excellent color graphics to simulate the setting of the adventure. For example, *The Pawn*

(Firebird, $44.95) uses graphics to simulate the world of Kerovnia, a land from which you must escape. You must explore the land and interact with the characters you meet in order to gather the clues you need to escape.

FTL's **Sundog Frozen Legacy** ($39.95) uses 3-D graphics to illustrate a spaceship which you pilot to different cities in different worlds in different star systems. You inherited the spaceship from a rich uncle. At the same time, you inherited the task of replenishing the population of your colony. In order to do this, you must navigate space, gathering supplies and colonists, exploring all these worlds while avoiding pirates who try to steal your cargo.

Borrowed Time (Activision, $44.95) places you in the position of solving a mystery and preventing your own murder. In the opening scene you, detective Sam Harlow, are in your office. If you leave your office you run into some tough character who are out to get you. Another Activision program, **Hacker** ($49.95), begins with the screen displaying "Logon Please." You take it from there in this simulation of breaking security codes to get access to information.

Some adventure simulations are so popular there are sequels for avid players. The popularity of **Temple of Asphai**, a fantasy adventure with many dangers and magical characters, prompted its producers to come up with two other programs, **Upper Reaches of Asphai** and **The Curse of Ra**. All three games for the ST come packaged together as the **Temple of Asphai Trilogy** (Epyx, $34.95).

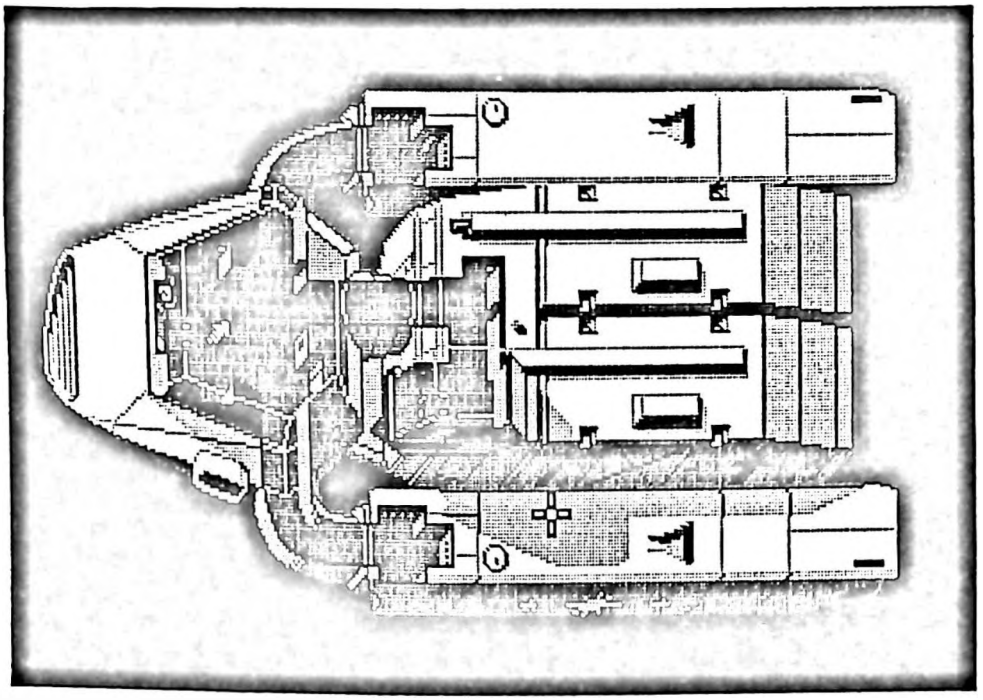


Figure 8.8-Interior of the Spaceship in Sundog.

Another program, **Transylvania** (Penguin Software, $39.95) takes you through adventures in the Royal Court of Wallachia. These adventures with King John the Good, Prince Erik, and Princess Sabrina are continued in Transylvania's sequel, **The Crimson Crown** ($39.95).

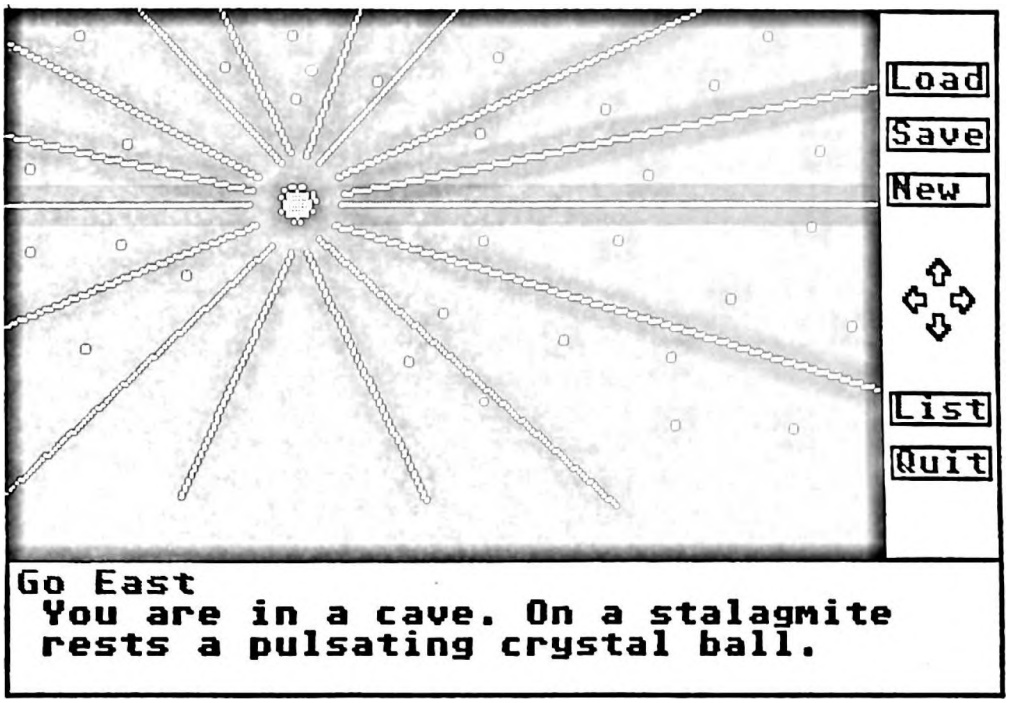


Figure 8.9-A Cave is one of the "rooms" in The Crimson Crown.

Even the youngsters can get into the simulation act. **Winnie the Pooh in the Hundred Acre Wood** ($24.95) is a combination of text and graphics in which all the famous characters appear.

Zork. The simulations discussed thus far rely heavily on color graphics. Other adventure simulations such as **Zork** (Infocom, $34.95) are a different type. They also have rooms, mysteries to be solved, and dangers to be avoided, but use words instead of color graphics to create the adventure. Zork is actually a series of adventures. **Zork III** ($39.95) puts you in a dark, foggy forest that is part of a complete Great Underground Empire.

Although the basic format of most adventures is similar, there are many variations on the basic theme. Infocom has a comprehensive collection of adventures for the ST. One, **Hitchhiker's Guide to the Galaxy** ($39.95), is based on a popular book. They also have mystery and science fiction simulations. **A Mind Forever Voyaging** ($44.95) takes place in the year 2031 and puts you in the role of an intelligent computer which can see into the future. And, one of Infocom's newest releases, **Ballyhoo** ($39.95), takes place in a circus environment. Many people feel the text adventures can create a richer environment for adventure because they tap the imagination of the mind.

Arcade Simulations

Games in this category are a sharp contrast to adventure simulations. Arcade simulations, which are also called "action computer games" or simply "arcade games" place emphasis on the computer's sophisticated animated graphics and sound effects. They are similar to the arcade games that earned millions of dollars, a quarter at a time, for their developers.

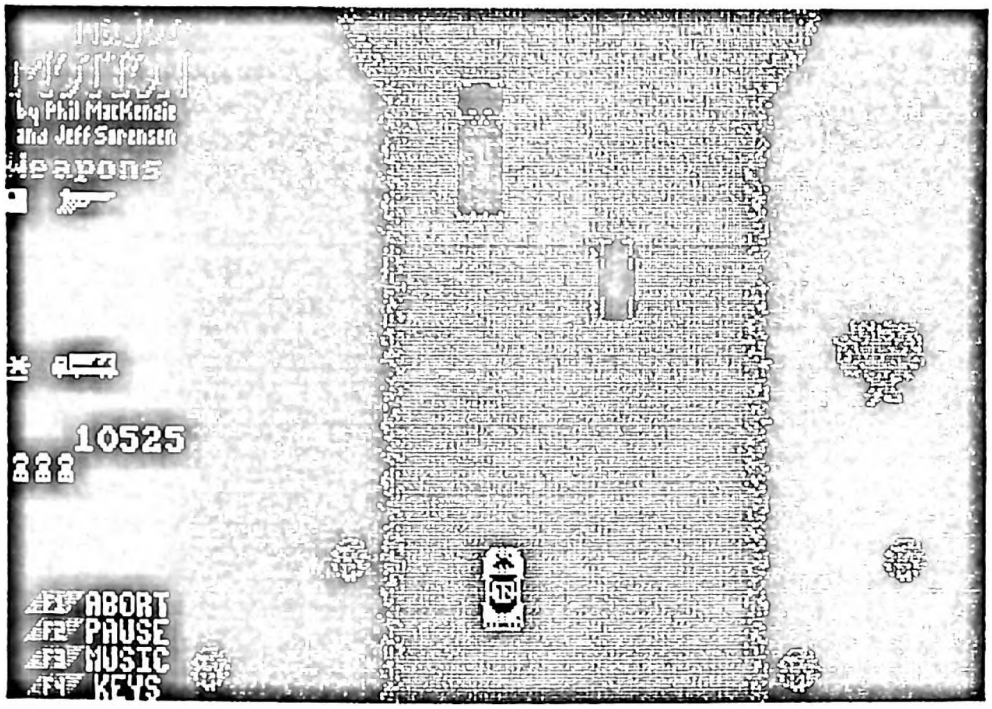


Figure 8.10-On the Road in Major Motion.

Most arcade simulations, like popular video games, are "real time games." That is, things are happening on the screen whether you do anything or not. While adventure simulations call for careful thought, analysis, and decision making, arcade simulations rely mostly on action. Doing nothing is usually disastrous. Quick responses are highly rewarded. Adventure simulations may take hours to learn; rules in arcade simulations are usually learned very quickly.

Major Motion (Michtron, $39.95). In this simulation you are a race-car driver commissioned to rid the highways of "enemy" cars from the Draconian League which will try to make your car run off the road and crash. You must either avoid these cars or use your weapons to get them off the highway. Your car is equipped with machine guns, but the "weapons truck" will deliver other weapons--missiles, smoke screens, oil slicks--as you travel along the highway. Points accumulate as you successfully maneuver down the highway eliminating League vehicles.

Time Bandit (MichTron, $39.95). Combining arcade action with some adventure simulation type interaction, Time Bandit is composed of a variety of "lands," each with sixteen different levels of activity. In each land you must gather all the available treasures you see as you explore while warding off numerous creatures which attack from all sides. In some lands you must solve mysteries. You are asked to answer a question or take some action. These parts of Time Bandit are somewhat like adventure simulations since text is displayed on the screen and you must type in responses. Success in this game depends on your finding all the treasures in each level of each land.

Flight Simulator (Sublogic, $49.95) and **Jet** ($39.95). Sublogic's Flight Simulator, which has been a popular program on many other computers, is also being released for the ST. If you have always wanted to fly a plane, this flight simulator is probably the next best thing to being there. You can choose to fly a Cessna 182--in Flight Simulator--or a Lear Jet--in Jet--using either the mouse or the keyboard for controls. The ST's windows allow you to see two different views from

the plane, watch yourself from the control tower, or watch from another plane. You can adjust the time, season, or flying conditions to obtain a variety of different challenging situations for your flight from Boston to New York.

Mean 18 (Activision, $49.95). The Bush Hills Country Club course is included on the Program Disk. If you want to try other courses, a separate Course Disk provides simulations of the famous courses at Pebble Beach, Augusta, and St. Andrews. Or, you can use the course architecture program to change as many holes as you wish on the Bush Hills course by adjusting the fairway and rough areas. The architecture program even lets you add or move sand traps, different kinds of trees, and water hazards.

Mean 18 has excellent graphics and sound effects. It is designed for one to four players and provides for both beginner and expert levels of difficulty and offers match or stroke play when more than one player is on the course. The documentation is not lengthy but provides a thorough explanation of the program. With a little practice you can control every aspect of your swing from height of backswing to club position upon ball impact. You can even work on your swing at a practice tee. Or, practice reading putts at a practice green.

Other Arcade Games

Magical spells, unicorns, and opponents such as Fitzroy the Triceratops, Manowa the jellyfish, and Amy the sentient mushroom are all a part of **Hex** (Mark of the Unicorn, $39.95). You must move around a grid, changing the colors of the hexagons which make up the grid before your opponent does. If you succeed, you advance to the next level and a more powerful opponent.

Rogue (Epyx, $34.95)) is an arcade game which is billed as a strategy game, which is never the same game twice. Most instructions which come with adventure simulations suggest that you make a map of your travels. Mapping doesn't help you with Rogue, however, since the rooms may change with each game.

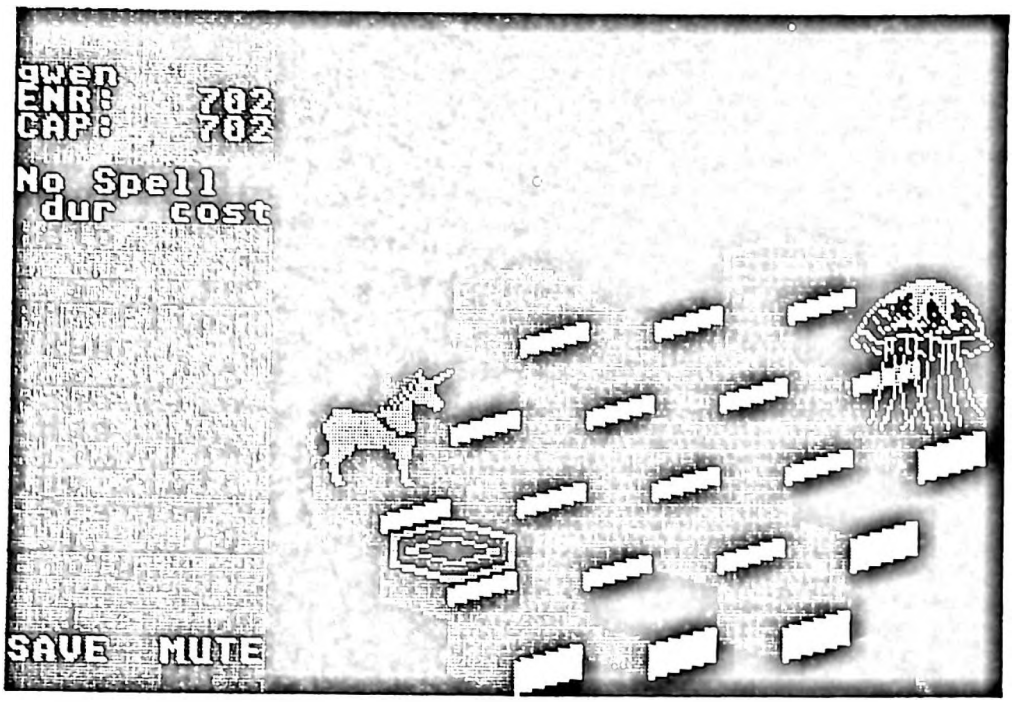


Figure 8.11-The Unicorn against Manowa in Hex from Mark of the Unicorn.

EDUCATIONAL COMPUTING

Using computers as a educational tool may involve the use of programs such as word processors, databases, and electronic spreadsheets; or, it may mean using the ST to teach computer literacy. Teachers might even use computerized gradebook programs. However, there are other programs which are designed with teaching in mind.

COMPUTER ASSISTED INSTRUCTION

The terms Computer Assisted Instruction (CAI), Computer Aided Learning (CAL), and Computer Aided Instruction (CAI) are all used to describe using a computer educationally. We will use the abbreviation CAI to refer to three types of computer based teaching approaches--drill and practice, tutorial, and simulations.

Drill and Practice

This simplest type of computer-assisted instruction doesn't really teach you anything. It just gives you practice in something you learned in some other way. Many programs, for example, give the student practice on basic math skills. The computer generates a problem and asks the student to type in the answer. The computer checks the answer and tells the student whether the answer was correct or incorrect.

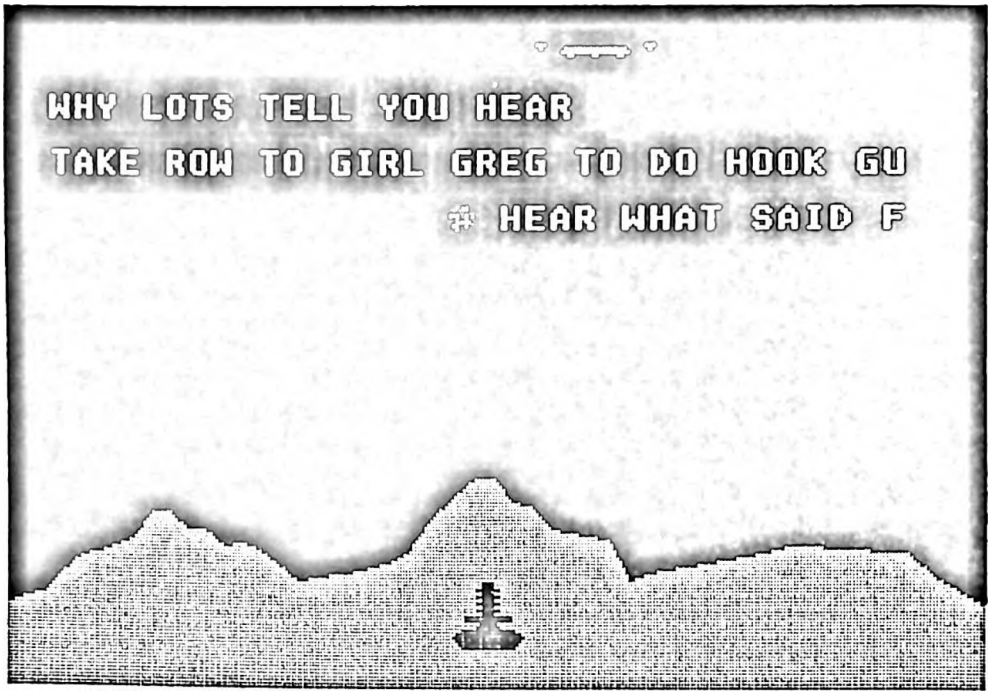


Figure 8.12-Shoot down the Word Invaders by typing the correct letters.

A more sophisticated type of drill and practice program keeps track of the student's errors. Errors are explained and help is provided. New problems are adjusted to the student's current needs. Some drill and practice programs select the difficulty level of problems on the basis of the student's performance during

earlier sessions. They remember errors and give the student extra practice in particularly troublesome areas.

Typing Tutor and **Word Invaders**. Academy Software's Typing Tutor ($24.95) is a very good drill and practice program. It is packaged with Word Invaders, an educational arcade game. Typing Tutor helps you practice touch-typing. It has eight levels of difficulty and is a colorful, well-written program. One review called it "The best typing tutor we have seen yet." The program displays a set of letter exercises on the screen. Then you type in the letters while the computer keeps track of your speed and accuracy. Each level provides a slightly different type of practice.

We recommend it for anyone who would like to improve touch-typing skills. The program keeps track of your errors as well as typing speed, and it highlights in red all the letters typed incorrectly. Making fewer than five mistakes allows you to advance to the next difficulty level.

Word Invaders, the second program in the package, is an excellent game that lets you practice your typing skills in a video game environment. Lines of words come marching down the screen. You must type them correctly to erase them from the screen. If any words reach the bottom of the screen you lose. The program assigns a score based on your typing speed and accuracy, assuming you are not destroyed by words that reach the bottom of the screen. This is an excellent program that makes typing practice enjoyable instead of drudgery.

Tutorial Programs

A tutorial program does more than provide practice on something you already know. It actually assumes some of the responsibility for teaching a skill. Tutorials are used in areas ranging from science subjects such as physics, chemistry, and biology, to programs for young children that teach shape, number, and letter recognition. High school students may find tutorials designed to help them prepare for taking the Scholastic Aptitude Test helpful.

Donald's Playground (Sierra On-Line, $29.95). Counting money and making change could be more fun for children if they learn the skills on Donald Duck's playground. In this tutorial Donald is the teacher.

Mathmagics. Volume 1 of this series of four from Eclipse Engineering deals with addition and subtraction. Other volumes include multiplication and division, fractions, and algebra. Each Volume sells for $39.95. The graphics used in these programs not only help teach the concepts but make learning more fun.

Simulations

Educational simulations, like adventure and arcade simulations, let you take a role in a particular environment. You may be the operator of a nuclear power plant or in a covered wagon on the Oregon Trail. The actions you take and decisions you make determine what happens next in the simulations. Simulations offer a method of studying some subjects which might otherwise not be covered because they are too dangerous, too complicated, too time-consuming, or too costly.

ST Planetarium (Atari, $29.95). As you might expect, this program simulates a planetarium in which you can view the sky from anywhere on earth in any time period. You can even print out information about a specific star you are interested in.

Logo

Another educational application has to do with a popular computer language, Logo. This language is included on your ST Language Disk. Chapters 13 and 14 cover Logo programming commands and techniques.

This chapter is not an exhaustive source of descriptions of home, recreational and educational software available for the ST. *ST Software Catalog* from Atari ($9.95) includes many other programs available. New editions of this catalog are published periodically in order to help keep you up to date. And, of course, we recommend reading monthly publications such as those listed in Chapter 16 as one good way to keep up with new software releases.

9

Sound and Graphics on the ST

Most people don't think of their computer as a musical instrument or an artist's canvas. People associate numbers, mathematics and business applications with computers, but a growing number of computer owners are using computers for artistic applications. The excellent sound and graphics capabilities of the ST are one of the major features which make this computer so popular. Some professional musicians, including Herbie Hancock, count personal computers in their complement of instruments. Computers have also been used in the show business industry to create special effects in movies like Tron, Star Wars, and War Games.

The artistic potential of the personal computer is also apparent in the color graphics, animation, and sound effects in popular computer games. Designing the kind of graphics you see in video games requires considerable computer knowledge and more than a dash of creativity, but there are other ways to explore the artistic potential of the ST. You can create some fantastic visual graphics and music without spending months learning to program.

You can harness the artistic talent of the ST in many different ways. With appropriate accessories and software, the ST becomes a creative tool. You can use it to compose music, draw artwork, or create animated graphics. Once created, the computer can be used as a medium of expression. RGB monitors can produce outstanding graphics even in the ST's low resolution mode, color plotters and printers--like the Okimate 20--can create printed copies of your work, and with music synthesizers you can play high-quality music.

The following sections describe some of the programs available for the ST. Hardware accessories mentioned in this chapter are discussed more thoroughly in the I/O and Accessories chapter.

GRAPHICS SOFTWARE

Outstanding original artwork depends primarily on the creativity and talent of the artist. Many of us have always been disappointed in our efforts to produce freehand artwork. Anyone can make a circle by drawing around a quarter or a square by drawing around a box, and most of us can draw a straight line with a ruler. But these tools are not very sophisticated drawing aids. The finished product will still depend on our own talent and creativity. Using a computer as a

drawing tool puts artistic success within the reach of all of us. Sophisticated graphics, detailed drawings, and intricate art and poster work can be created with many of the programs available today. Some programs even have electronic drawing aids or a disk library of pictures you can combine with your own artwork or use alone.

A drawing toy, Magic Slate, has been popular with children for many years. The Magic Slate is inexpensive and consists of nothing more than a drawing surface made of a thin sheet of plastic attached to a rather waxy backing material and a wooden stylus shaped like a sharpened pencil. The child uses the stylus to draw on the plastic and whatever is drawn appears on the light gray drawing surface as a dark gray picture or design. However, any drawings produced are lost when the plastic drawing surface is lifted from the background. Although this makes the slate easily reusable, nothing can be displayed more than once without being redrawn.

One type of accessory that has been a very popular drawing tool for some computers is the graphics pad. The graphics pad is a sophisticated electronic version of the Magic Slate with a drawing surface on which you draw with a stylus. But it offers advantages over the toy because artwork can be produced in color and can be saved for redisplaying as many times as you wish. Graphics pads allow you to draw freehand but they also have options which let you push a button and produce a perfect circle or square in any size and at any location on the screen. Some have a variety of brush sizes which allowing you to vary the width and style of your lines. You may even be able to select a background design. If you wish, you can use all the features of the graphics pad to compose a picture rather than draw one.

Several software programs exist which give you and your ST all the capabilities of a graphics pad--and more--without the device itself. These programs use the mouse for a stylus so no special accessories are necessary.

NEOchrome. The NEOchrome program is one example of a graphics program for the Atari ST. One version of this program, NEOchrome Sampler, is included on the language disk which came with your computer. Atari is continuously releasing updated versions of this program so the version you have may differ slightly from the one described here. The basic functions and capabilities should remain the same, however.

When NEOchrome is loaded, the bottom half of the screen displays the menu. Across the top of the menu is the color bar, or palette. On the left side is a grid which shows icons representing the drawing tools and some of the other NEOchrome functions. The center portion of the screen is another grid showing a rainbow of available drawing or painting colors. The right side of the screen shows the color mixer, the UNDO box, and a blank area which is used to display options when certain tools are selected.

You choose a color by clicking on the color palette or by using the color mixer, then draw pictures on the top half of the split screen or on the Full Screen. A small white triangle is visible below the palette box containing the current color selection. If you look at the rainbow grid at the bottom of the screen you will notice a box around the center square of the grid. You can move this highlighter box to any other square on the grid, double-click the mouse button, and that color will appear on the color palette in the position marked by the triangle. There are 512 colors available. The color palette displays 16 colors-- the maximum number you can use in a picture.

Freehand drawing is accomplished by selecting the pencil icon, moving the pointer to the drawing area, and moving the mouse to draw lines. When drawing on a split screen you can select the grabber icon--represented by a hand on the

grid--and use it to scroll the drawing area up and down. Selecting the paintbrush or spray can displays all the possible paint patterns. Choose one and use the mouse the same way you did with the pencil. Try experimenting with different brush sizes and patterns to see the various effects. If something doesn't look just right, select the eraser icon and the mouse acts like an eraser when you move it around on your picture. Clicking twice on the eraser clears the whole picture.

To put text in your pictures, select the T icon, move the cursor to the point on the screen where you want the text to appear and begin typing. The lower right corner of the screen displays the bold, italic, and font options available when using text.

The paint can icon is used to fill areas of your picture with solid color. The icon which looks like a box made with dashed lines is the copy box. You can put the box around a section of your picture and move that section somewhere else in the picture.

You can save the drawing to disk by choosing the disk icon and clicking the right mouse button then reload it by clicking the left mouse button. You may also creates special effects by using your drawings with the NEOchrome Slide Show program.

NEOchrome Companions. Other companies are releasing software which can be used with NEOchrome. The Dragon Group produces disk libraries of pictures which can be loaded as NEOchrome pictures. Once they are loaded, you can use all of NEOchrome's tools to make changes and save your version. These disks are a part of the ST Coloring Book series. The first release in the series, *The Sampler I* ($34.95), contains pictures of animals, birds, rockets, leaves, flowers, and a collection of borders. Another release, *The World*, contains line maps of the United States and other countries around the world.

Figure 9.1-United States Map from the ST Coloring Book.

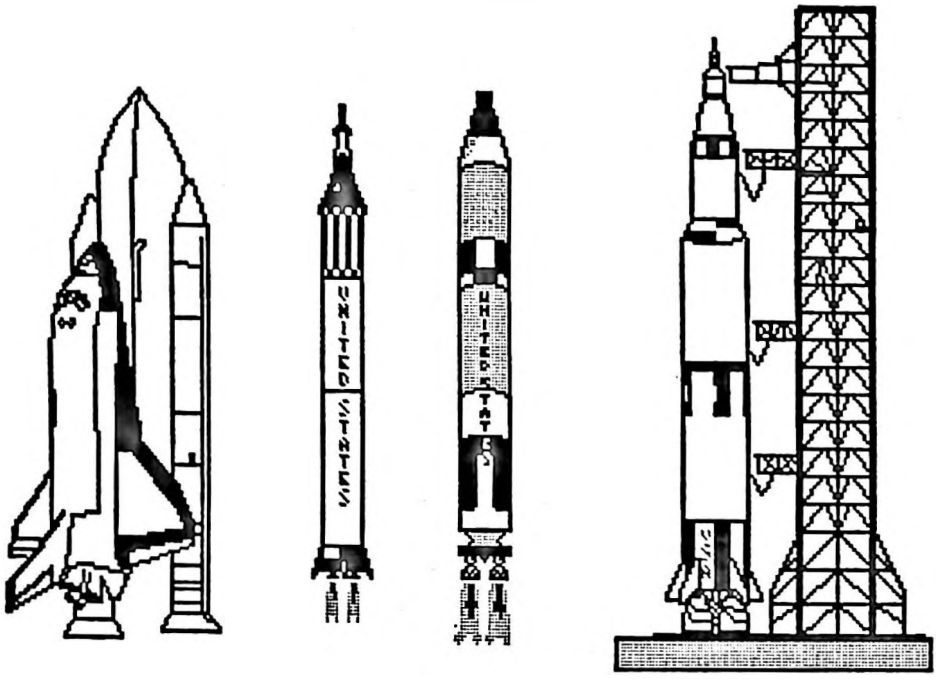


Figure 9.2-Space Vehicles from The Sampler, ST Coloring Book Series.

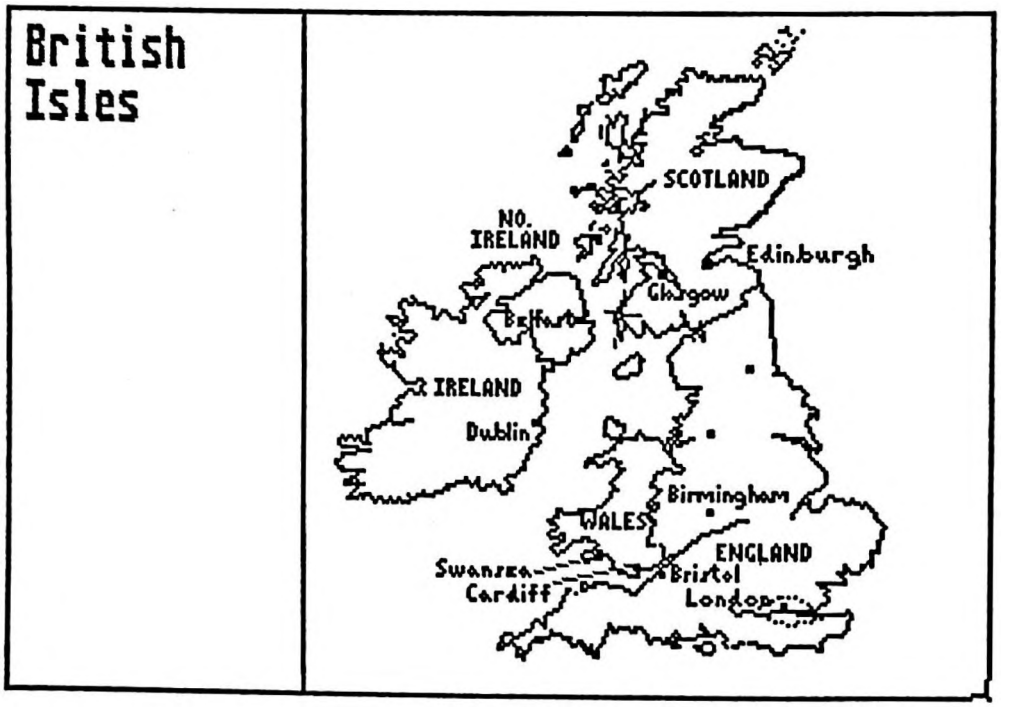


Figure 9.3-Sample Map from The World, ST Coloring Book Series.

Powervision (Power Systems, $24.95) also features NEOchrome pictures to develop slide show presentations. Up to 80 NEOchrome images can be used on one disk. A variety of special effects are available. Picture overlays can be designed so that the new picture appears from the left, right, top, or bottom of the screen. For instance, if you want to design a slide showing expenditures for the last year, you might combine several pictures. Design one picture for the lines on the graph, another for the graph labels, another for the bars on the graph, and still another for the caption line to be placed under the graph. When all the pictures are overlaid, your slide is complete. This program produces some very professional images which could easily compete with some of those you see on television.

DEGAS. If you have looked at any of the monthly publications containing Atari articles in recent months, you have probably seen some examples of DEGAS graphics. *DEGAS* (Batteries Included, $39.95) is the Design and Entertainment Graphic Arts System. Several of the magazines have held graphics contests and many of the winning pictures were designed with this program. At the time of this writing, DEGAS is, by far, the most sophisticated graphics program available for the ST.

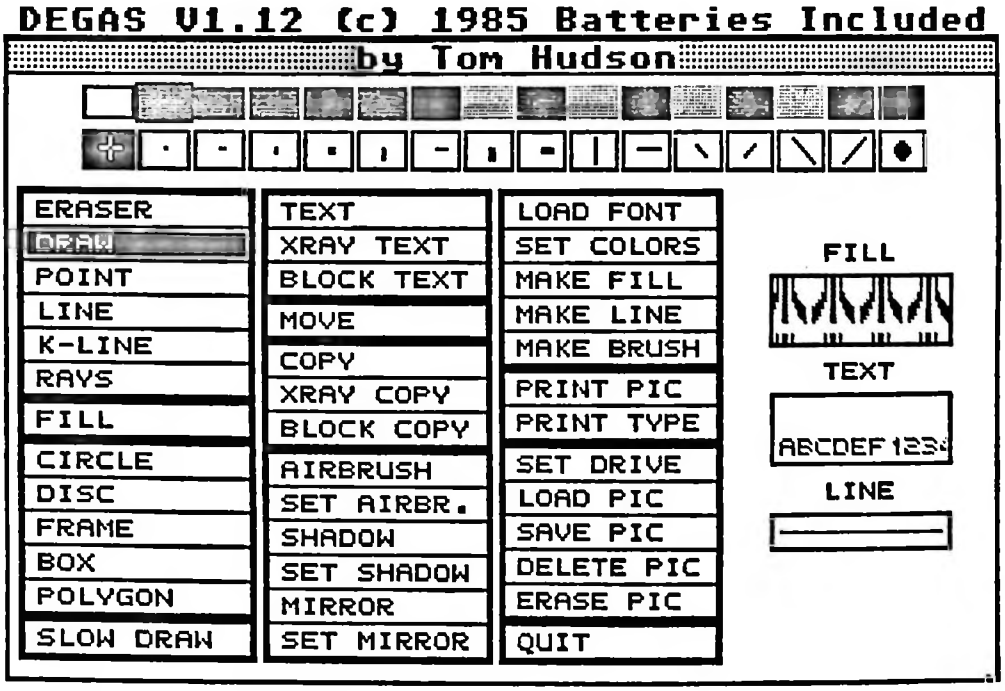


Figure 9.4-The DEGAS Main Menu.

The DEGAS main menu screen shows 39 different options, 16 colors and brush sizes on two bars across the top of the screen, and boxes which display the current fill pattern, text font, and line style. Menu options allow you to draw freehand, load text fonts, and move or copy portions of your picture. You can also load, save, print, or erase pictures.

The following are other available menu options:

Point--leaves a single brush image on the drawing screen instead of drawing lines.

Line--draws a straight line from one designated point to another.

K-lines--draws continuous lines. The beginning point of one line is the ending point of the previous line. Move the cursor to establish line placement, then click the mouse button to set the line in place.

Rays--draws straight lines which radiate from the same start point.

Circle and Disc--draw circles with the mouse by establishing a center point and expanding from that point to the desired size. The Disc option draws the same figure as Circle but the figure is filled with the current Fill pattern.

Frame and Box--works like Circle and Disc except these draw squares and filled boxes.

Polygon--draws irregular, filled shapes using the continuous line method described in K-lines.

If that isn't enough options, DEGAS has some other very interesting features. You can Fill any section of a picture you draw by choosing that option on the menu. There are 38 built-in fill patterns or you can design your own. To change the Fill pattern, position the cursor on the Fill box and click the left mouse button. You can use this procedure to preview all of the preset patterns. When using the Fill option, the section you want filled must be closed. If it is not, the pattern will leak out and cover areas you didn't intend to fill.

If you want to paint an area of the screen without making it a solid color, use the Airbrush option. The result is lines which appear to have been painted with a spray can. Use the airbrush to add shading to three-dimensional pictures. Airbrush works only in the freehand drawing mode. You can even customize the size and flow of the airbrush.

Figure 9.5-Design of the name "Sheryn" mirrored in three directions.

Shadow duplicates any line or figure drawn. The Set Shadow option allows you to specify which direction from the original point or line the shadow will appear and the distance--in pixels--the shadow will be from the original point. As shown in Figure 9.5, you may achieve some unexpected results by using the Mirror option. The mirror can be vertical, horizontal, diagonal or in all three directions.

MichTron's **Animator** ($49.95) produces animation with pictures designed in either the NEOchrome or DEGAS program. In order to animate your pictures, you must design the background, draw the series of pictures to be used, and develop "masks" which the program uses to organize the pictures into their animated sequence.

Easy-Draw (Migraph, $149.95) is a GEM-based program designed primarily for business applications. The disk contains four example pictures: a floor plan, geometric figures, business forms, and an organization chart. The desktop not only features drop-down menus on the menu bar across the top of the screen but also has a series of pop-up drawing menus.

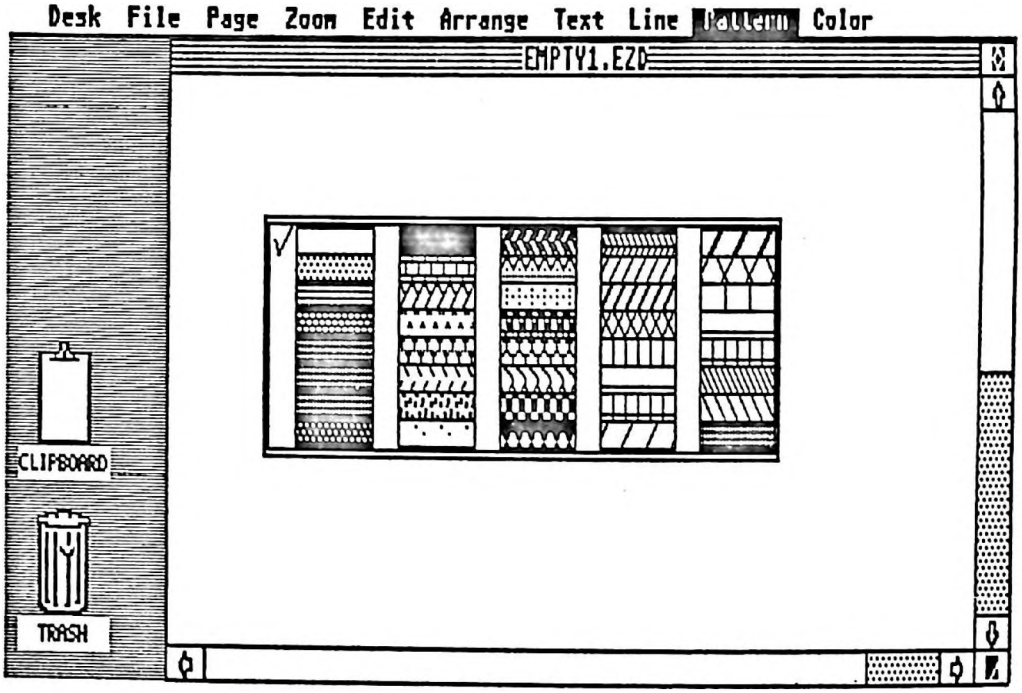


Figure 9.6-The Easy-Draw screen with the Fill Pattern menu visible.

Some of the Easy-Draw features are similar to the other programs previously described. The program will draw circles, squares, lines, arcs, connected lines, ellipses and rounded squares. The figures can be "cut" by transferring them to a temporary storage area and "pasted" to other figures to make drawings. Almost forty fill patterns are available along with the option to design your own pattern.

Easy-Draw's manual is very well-written and thorough. Step by step directions are given for using the menus and carrying out all the commands available in the program. It even includes detailed instructions are provided for each of the sample pictures included.

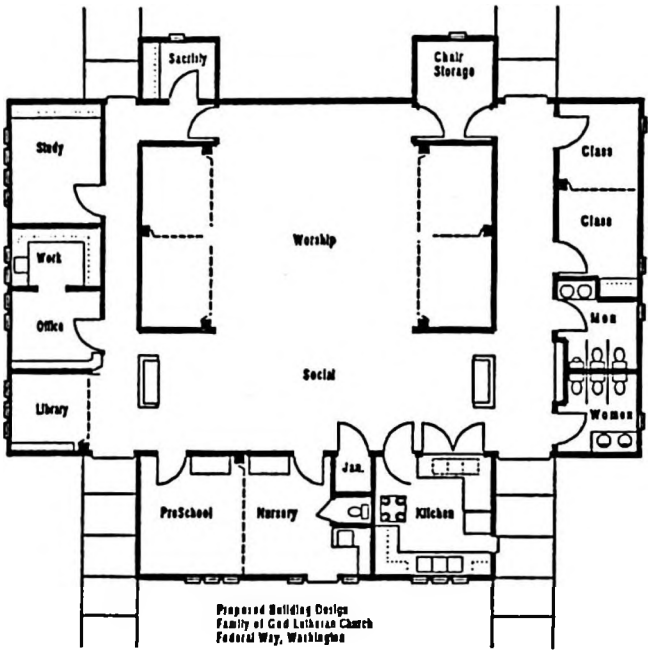


Figure 9.7-Architect's floor plan drawn with Easy-Draw. (Courtesy of MIGRAPH, Inc. by Michael Longe')

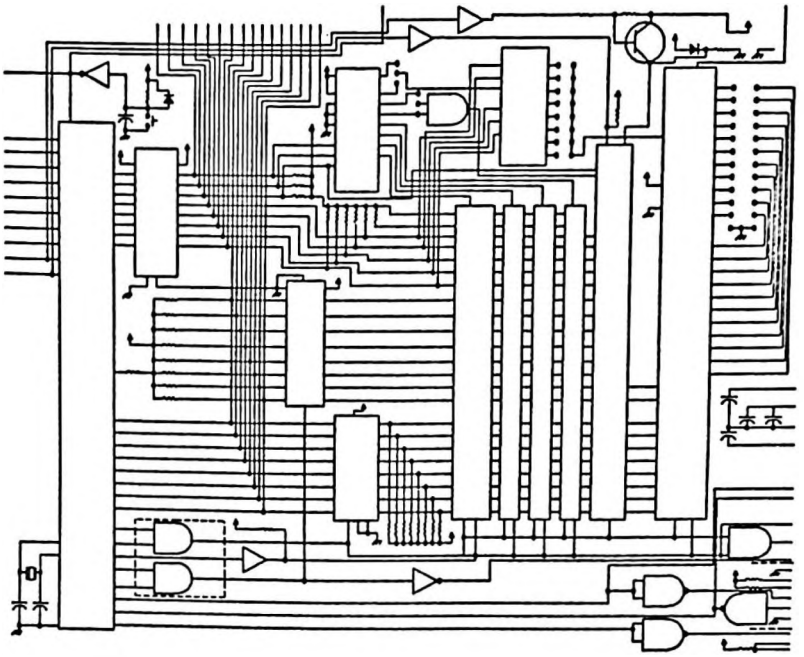


Figure 9.8-Schematic diagram drawn with Easy-Draw. (Courtesy of MIGRAPH, Inc. by Michael Longe')

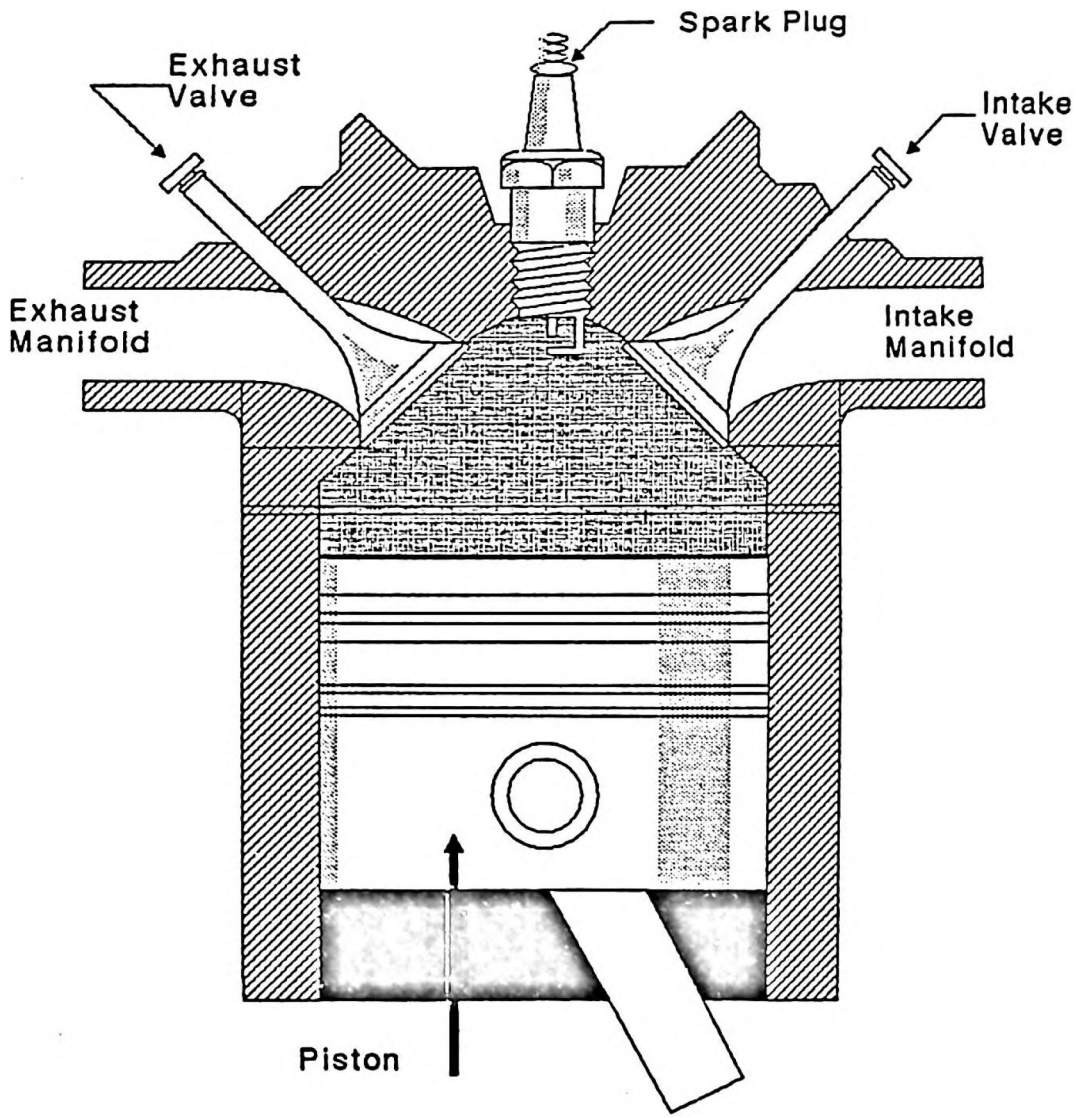


Figure 9.9-Cutaway of a piston drawn with Easy-Draw. (Courtesy of MICRAPH, Inc.)

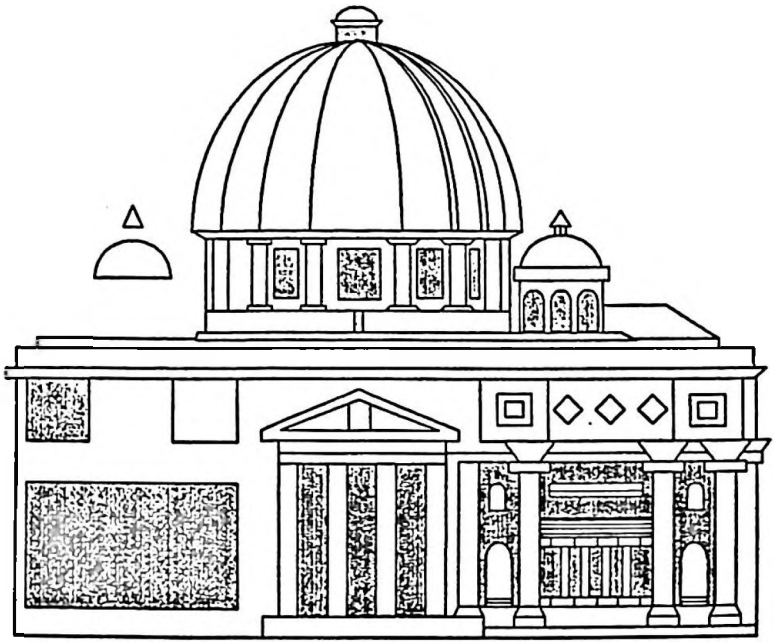


Figure 9.10-Building design drawn with Easy-Draw. (Courtesy of MIGRAPH, Inc.)

CAD 3-D. This program from Antic Software was designed by the author of DE-GAS, Tom Hudson, but its purpose is quite different. *CAD 3-D* ($49.95) is another program which could easily find a home in the business world. CAD stands for Computer Aided Design. Such programs are used by industrial and mechanical engineers, architects, and other professionals who need quality drawings and designs. When automakers are designing new styles of cars, they use sophisticated CAD programs to create wire-frame forms of the car's shape long before the first car rolls off the assembly line. A city in Texas recently used a CAD program to design a new flood control system.

With this program, wire forms or solid representations of objects are designed as though you were actually using machines like lathes and jigsaws. The design can be rotated for examination from any viewpoint. The program also has a "light source" feature which can also be moved to different locations to change perspectives.

The Graphic Artist (Progressive Computer Applications, $199.00) was originally marketed as a combination graphics program, word processor, font editor, and electronic spreadsheet selling for $495. To make their program more attractive to the home user, the company has now begun marketing the graphics program and font editor ($79.00) separately. At this writing this text editor was the only program available for the ST which outputs to alaser printer.

Data for producing the designs and drawings is stored in the built-in spreadsheet as commands, and fixed coordinate pairs make the editing process much easier since all you need to do to change part of the design is change the information in the spreadsheet cells. This also allows for seeing what difference minor changes in the coordinates would make in the overall design. Combined with the text editor this program has the potential to produce professional quality documents and graphics for use in presentations or business communications.

Graphic Artist Quarterly

Vol. 1 - No. 1 May 1986

Computer Aided Design Plus Desktop Publishing

If you have been looking for a Computer Aided Design (CAD) system that you could use for Desktop Publishing, your search can stop right here. Progressive Computer Applications, Inc. (PCA, Inc.) recently released their new product, The Graphic Artist. Functionally equivalent just in the CAD capabilities to the very popular CAD program "AutoCAD" (available on the IBM PC), The Graphic Artist goes beyond just CAD. It also includes Desktop Publishing features in the same environment! Features such as multiple fonts, software bold, italics, underline and more provide the capabilities to create professional quality output.

The Graphic Artist runs on the exciting new Atari ST machine from Atari, Inc. The Graphic Artist combined with an Atari 520ST provides a CAD workstation that has a retail price of under $1300, far less than AutoCAD on an IBM PC!

Graphic Artist Feature Highlights

CAD Highlights

- Vector graphics
- Symbol libraries
- Precision drawing to exact scale
- Rotate to any degree
- Zoom
- Pan

.. and many others

DP Highlights

- Multiple font styles in-line
- All fonts in any size
- Snaking columns
- Word processing with auto word wrap
- Bold, *italics*, underline
- Create your own fonts

.. and more ....

Additional Highlights

- Output to dot-matrix printers, laser printers, color plotters
- Operates in monochrome and color
- Includes a spreadsheet for data storage and manipulation
- Includes business graphics

(Output on an HP LaserJet+ at 150 dots/inch)

Figure 9.11-Sample document produced with Graphic Artist and HP LaserJet+ Printer. (Courtesy of Progressive Computer Applications, Inc.)

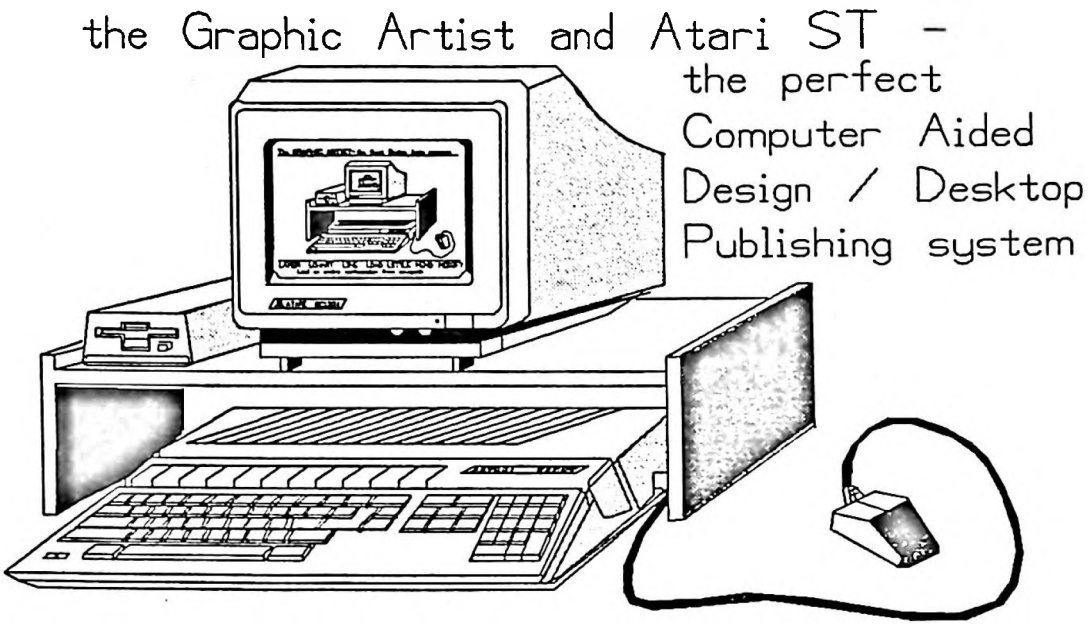


Figure 9.12-Atari ST Computer drawn with Graphic Artist. (Courtesy of Progressive Computer Applications, Inc.)

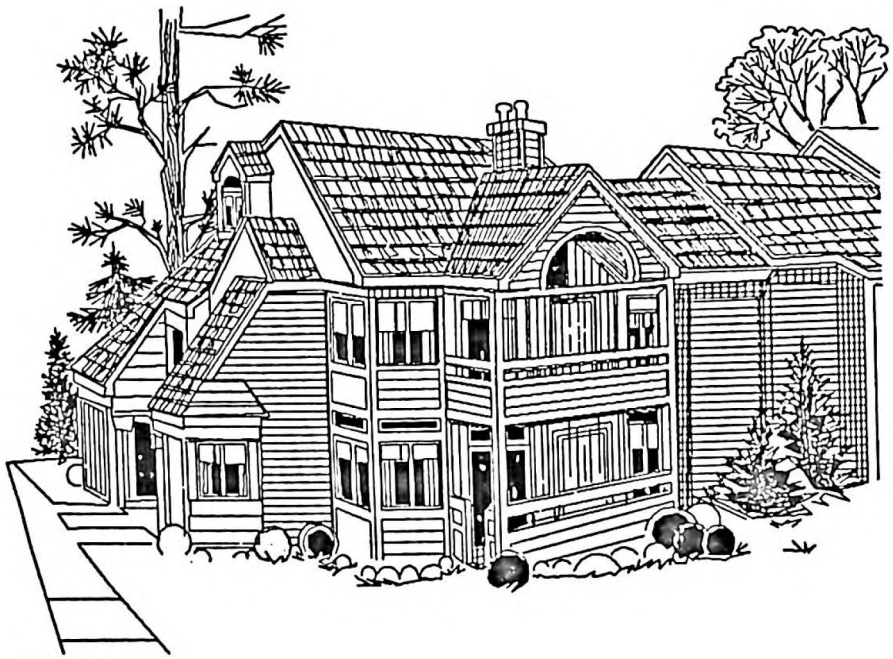


Figure 9.13-Apartments drawn with Graphic Artist. (Courtesy of Progressive Computer Applications, Inc.)

Typesetter ST. XLent Software's Typesetter ST ($39.95) also allows you to create a professional-looking document combining text and graphics. You can create church newsletters, business documents, a school paper, or a fancy letter to send to friends.

A companion program, *Rubber Stamp ST* ($39.95), allows you to make DEGAS and NEOchrome pictures into icons that can be used in Typesetter ST, documents. If you are creating a newsletter and need some artwork, just fire up your ST, draw a NEOchrome or Degas picture, and transfer it.

Printmaster. This program is one of the most useful pieces of software ever developed! *Printmaster* (Unison World, $39.95) allows you to create greeting cards, banners, and signs. You can even develop your own letterhead stationery and you don't have to know anything about computer graphics or programming to achieve very nice looking products. The program can pay for itself if you use it to make birthday and Christmas cards instead of buying them.

The Printmaster manual is around 30 pages long but is very complete. In fact, this program is menu driven and so easy to use you won't even need the documentation for most tasks. You can use any of 11 different borders and 8 different styles of type in your creations. The program disk contains over 100 drawings. You can select any of the assortment of drawings that fit holiday seasons such as Christmas, Valentine's Day, Easter, or Halloween. There is also a wide selection of other drawings ranging from dogs, cats, teddy bears, and penguins to houses, cars, and a UFO. You can use the drawings as they are, change them with the Graphics Editor within the Printmaster, or create your own with the Graphics Editor.

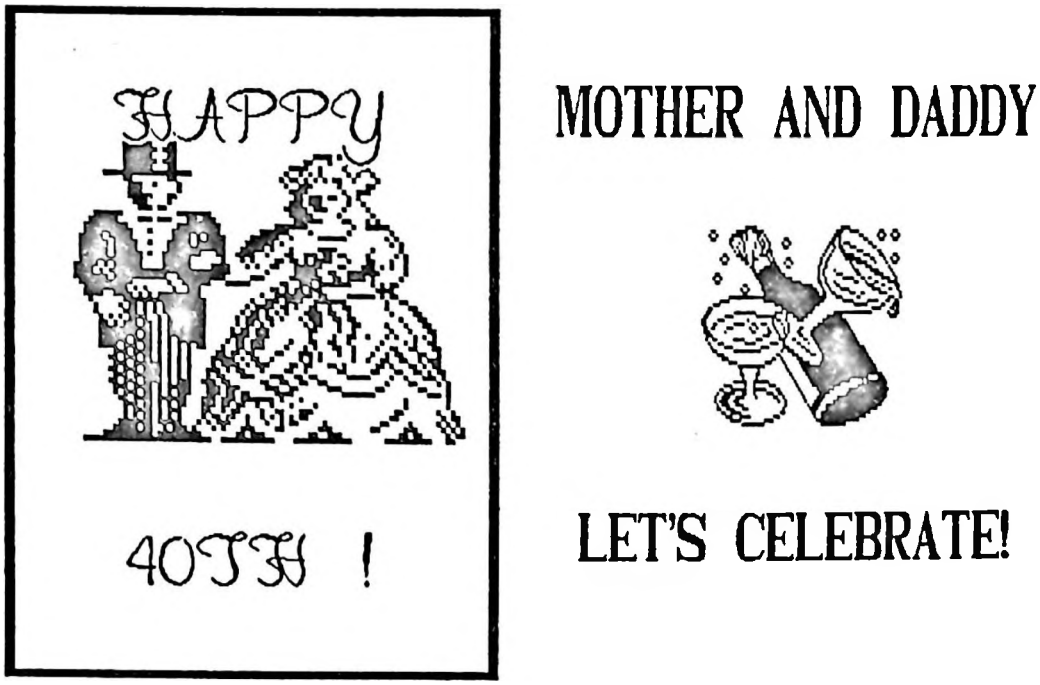
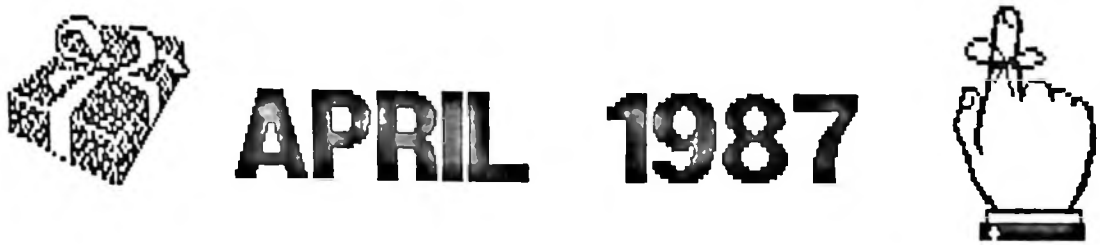


Figure 9.14-Front and inside views of greeting card designed with Printmaster.

You can use two different graphics and more than one type style on greeting cards or banners. The program also has a calendar option with which you can print monthly or weekly activity calendars. All you need to do is tell Printmaster



SUN	MON	TUE	WED	THU	FRI	SAT
			1 KRIS'S B.DAY	2	3	4
5	6	7	8	9	10	11
12	13	14	15 TAXES DUE	16	17	18
19	20	21 KEVIN B.DAY	22	23	24	25
26	27	28	29	30		

Figure 9.15-Calendar designed with Printmaster.

the month and year and it will print out the calendar for any month in any year between 1900 and 2099.

Unison World has also released even more graphics for use with Printmaster. *Art Gallery I* ($39.95) is a graphics library disk with 140 different pictures.

Video Digitizers. *The Hippovision Video Digitizer* ($139.95) is a hardware/software package that lets you capture video images and save them as graphics images the ST can display and print. The digitizer can capture images from a video camera, a VHS or Beta video tape, a television with a composite video output, or another computer. The digitizer produces black and white graphics (color is converted to 16 shades of gray) with a resolution of 320 by 250. The hardware plugs into the cartridge slot on the left side of the ST. This digitizer generates graphics files that are compatible with both NEOchrome and DEGAS graphics programs. Once the image is in a file you can use a program like DEGAS to add color to it.

With this software and a similar program from Digital Vision, ComputerEyes ($129.95), you could use a video camera to input the image of a picture from a magazine (or capture a picture from commercial television or cable) and then modify and edit the image. ComputerEyes images can be used as T-shirt transfers. HippoVision Video Digitizer images can be used to create rudimentary animation by displaying a sequence of images on the screen.

Figure 9.16-Sample digitized cartoon picture from ComputerEyes. (Courtesy of Digital Vision)

Figure 9.17-Sample picture from ComputerEyes. (Courtesy of Digital Vision)

MUSIC SOFTWARE

The use of electronic equipment such as computers and synthesizers is a developing phenomenon among both amateur and professional musicians. Professionals such as Wendy Carlos and Frank Zappa are only two of the many who use a computer as a musical instrument. High school stage bands and even some churches are now using electronic keyboards.

The Music Studio ($59.95) from Activision is a program for both beginners and advanced musicians. As DEGAS is probably the most popular graphics program for the ST, Music Studio appears to win the music software popularity contest hands-down. Perhaps that is because it is the only program currently available for the ST which allows you to compose everything from a simple version of "Mary Had A Little Lamb" to a complicated arrangement of "Beethoven's Moonlight Sonata" with nothing but your computer.

If you have a modem, logon to CompuServe and check the Atari 16-bit Special Interest Group forum. There are lots of music program files that can be downloaded and used with Music Studio. This could turn out to be one of the best sources of information about music on your ST.

MIDI Interfaces and Software

MIDI stands for Musical Instrument Digital Interface. It is a system for connecting electronic music instruments to each other. MIDI was originally developed to solve a major problem musicians faced when trying to use more than one electronic music instrument. Hundreds of companies manufacture components

for electronic music equipment, but until recently most of them designed their products without considering whether they could be connected to equipment made by other manufacturers. The musician who happened to own equipment from more than one manufacturer often ran into any number of problems trying to interface the equipment. The incompatibility problem was addressed by Sequential Circuits, an American company, working with several Japanese companies. In 1983, these companies cooperated in developing a standardized interface (MIDI).

MIDI itself is not a musical instrument, but it provides the necessary circuits for interconnecting a number of electronic music instruments. The Atari ST has a built-in MIDI port which makes it even easier to interface with electronic instruments. With MIDI you can control and communicate data to as many as sixteen different musical instruments. MIDI's standardized interface design allows you to turn notes off and on, indicate how hard and how quickly a key was played, select a preset sound pattern, and change other controls of an instrument depending on the instrument's capabilities.

All this information about what MIDI can do for professional musicians is interesting but you may be wondering what it has to do with you and your ST. MIDI offers tremendous potential for interfacing computers with electronic musical instruments and synthesizers.

Popular non-MIDI software like The Music Studio is excellent if you want to create music note-by-note on the screen and then play it through the computer's own music synthesizer. The advantage of MIDI comes from using the computer to control a musical instrument or group of instruments instead of having the computer play music for you. You could be an electronic one-man band. The sound capabilities of your computer may be great but they can't compare to the sound produced by the instruments themselves.

MIDI thus has considerable potential for increasing the music production possible with your computer, but this potential cannot be realized without appropriate software. Several companies have already begun to develop software that will allow you to use music synthesizers with your ST. These programs give you and your computer the potential to function as a miniature recording studio. You can record several music tracks and combine them. Add other devices such as electronic drums and the results can certainly be professional quality.

Hybrid Arts Editing Systems. Music synthesizers are programmable instruments by themselves, but with MIDI- compatible software you can control them with your computer. You use the software and computer as a composing, editing, and recording tool. Then you play your compositions through a MIDI compatible synthesizer such as the *Casio CZ 101*, *Yamaha DX* and *TX*, and *Ensoniq Mirage*. Hybrid Arts programs are either already available or under development for all of these popular synthesizers.

ST Sound Digitizer ($139.95). Hippopotamus also sells a sound digitizer for the ST. This device can accept sound input from a microphone, guitar, piano, electronic music synthesizer, or any other sound source, and store them in the memory of the computer. The stored sounds can then be played back through the device. The software supplied allows you to edit and modify the sound data in many ways. The sound is output through the Hippo Sound Digitizer. If you need additional amplification, the output can be sent to an external amplifier. Another program which is similar but less sophisticated is *Digisound ST* (Alpha Systems, $49.95).

The final music product for the ST we will discuss is a little different. QRS, an established company in Buffalo, New York, has been in business since 1900. The company has a huge library of player piano rolls of music from famous composers and musicians such as Scott Joplin, George Gershwin, Liberace, Bach, and Beethoven. QRS has converted thousands of their piano rolls to digital signals

and stored them on disks. With an electronic musical instrument that is MIDI compatible and QRS's *MIDI MAGIC* interface ($39.95), you can hear, and play along with, hundreds of compositions played as they were played by the musicians who made them famous. QRS products are distributed by Micro-W Distributing Company.

10

Programming in Basic

The next two chapters will help you learn to program in the popular computer language called BASIC. In this chapter you will be introduced to the ST BASIC environment, its menu bars and windows. You will also learn about BASIC commands, how to type in programs correctly, and how to edit them if you make mistakes. In the next chapter you will learn how to program graphics and sound in ST BASIC.

GETTING THE MOST FROM CHAPTERS 10 AND 11

These chapters are not intended to make you an expert computer programmer. They are designed to give you a good background in the concept of how to use a high-level computer language and what to look for in BASIC programs. Several books and magazine articles are already available on ATARI ST BASIC.

Monthly publications such as *Compute!*, *Analog*, and *Antic* often contain program listings you can type in and use. The following two chapters will give you enough background in BASIC to understand how such programs work and how to add simple enhancements to customize them for your own purposes. You will even know enough about the programs to debug them if they aren't working properly.

The best way to approach these chapters is to sit at your computer and type in the examples as you read. You could just read the chapters and try to remember the terms and instructions. However, learning a computer language is similar to learning a foreign language. You wouldn't try to learn Spanish simply by reading it. Learning Spanish, or any other language, is much easier if you practice it.

Practice is also the best way to learn a programming language. The examples in these chapters help you practice getting your computer to do what you want by programming it yourself. You don't need to memorize the definitions and instructions you see in these chapters. Once you have a little experience working with BASIC programs, the important points will become second nature to you.

INTRODUCTION TO BASIC

If you are from the United States, you may have some difficulty understanding someone from England or Australia. In fact, if you are from Boston you

may even have difficulty understanding someone from Lubbock, Texas. Even though two people speak the same language, sometimes the words they use may have different meanings or different sounds depending on where the person speaking comes from. That is, as you know, because of the different dialects of English. You could also encounter different dialects of Chinese if you travelled in that country.

Just as spoken languages have different forms, the BASIC programming language has different versions. All versions of BASIC are built around some of the same terms and concepts. But they usually vary somewhat from computer to computer. That is, Atari BASIC is not exactly the same as Commodore or Apple BASIC. In fact, there are several versions of BASIC available for your ST. One of them is included on the Language Disk which came with your computer when you purchased it. Others are available from third party developers. The examples in these chapters will use the version of BASIC which is on your Language Disk. They will be similar to programs written in other versions but may need slight modifications to execute properly.

BASIC is the most popular language for computers like the ST. That's why it is included on the Language Disk. BASIC allows you to communicate with the computer in an easy-to-learn language.

The process is much like talking to another human, but there are important differences. One major difference between talking to your computer and talking to a person is the need to give the computer exact instructions. In written and spoken English, you can mispronounce or misspell a word and still be understood. English is also a rich language, with many different words that mean the same thing. That is not the case in BASIC. The vocabulary of BASIC is limited, and you must use the right words exactly the way the computer expects you to use them.

Another difference between BASIC and English is the way punctuation marks are used. They take on added importance in BASIC. Rules for using commas, quotation marks, colons, and semicolons must be followed precisely. Omitting a single punctuation mark can prevent your program from running properly.

When the ST works in BASIC, it understands and assigns special meaning to certain words. These are called keywords. Different versions of BASIC use many of the same keywords, but there are differences in the words each version understands and, in some cases, the way the computer uses them.

THE ST BASIC ENVIRONMENT

Loading BASIC

All you need to do to begin using ST BASIC is to load it from the Language Disk into the memory of the computer. To do this, insert the TOS System Disk into the disk drive, turn on the drive and the computer. Once the desktop appears on the screen, remove the System Disk and insert the ST Language Disk in the drive. (If you have two disk drives, leave the System Disk in Drive A and insert the Language Disk in Drive B.) Move the mouse pointer and double-click on the Disk A icon in the upper left corner. When the files appear in the window, double-click on the BASIC.PRG icon.

Four Windows

Once ST BASIC is loaded, the screen will display a new menu bar across the top. The options on this menu bar will be covered later. The desktop area will change to display the BASIC desktop. The BASIC desktop is made up of four sepa-

rate windows: Command, Output, List, and Edit as shown in Figure 10.1. The Command, Output, and List windows are labeled. A brief description of each window follows. You will get some practice using the windows as you work through these chapters.

The Command window covers the lower portion of the desktop. When BASIC is loaded, the top edge of the Command window is shaded to indicate this window is activated. An OK prompt and a blinking square, called the cursor, appear in the bottom left corner of the Command window. The OK prompt means ST BASIC is waiting for you to begin typing in a program. As you work through the examples in these chapters, any commands and program lines you enter will appear in this window.

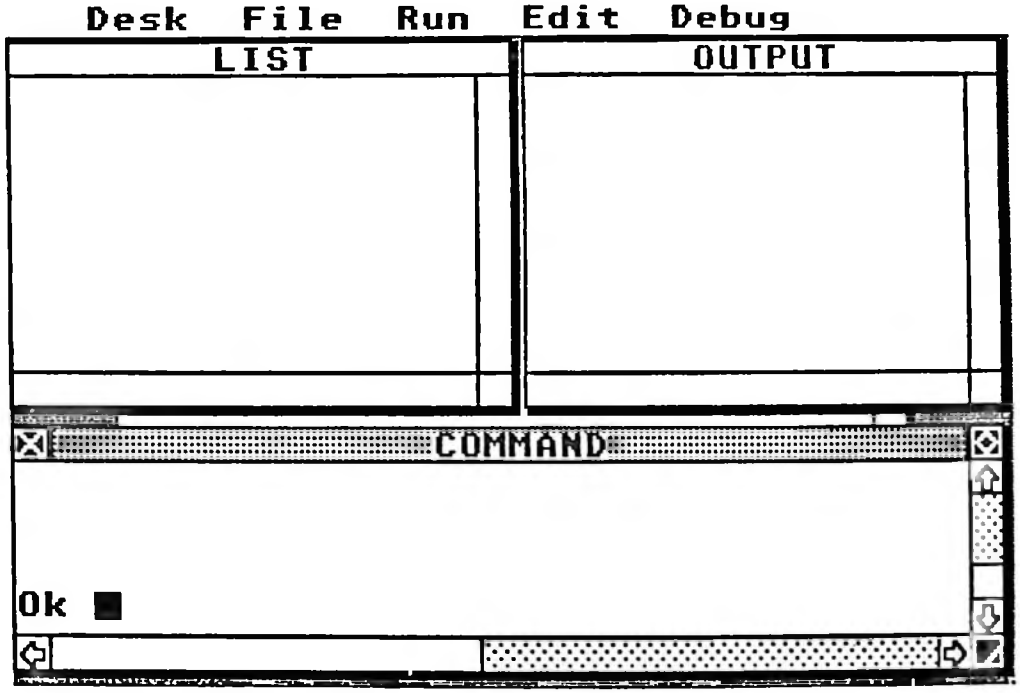


Figure 10.1-BASIC screen showing the four windows.

The Output window is on the right side of the upper half of the BASIC desktop. This section of the desktop is used to display the results when your program is RUN.

The List window is on the left side of the upper portion of the BASIC desktop. In BASIC, the keyword LIST tells the computer to display the lines of the program in the computer's memory. Any program listing will appear in this window.

The Edit window is barely visible on the BASIC desktop. One line of this window appears between the List and Output windows in the upper half and the Command window in the lower half of the Desktop. Actually, the windows appear to be "stacked" on the desktop and the Edit window is "under" the other three windows. If you close one of the other three windows, more of the Edit window is visible. If you close the Command, Output, and List windows the Edit window is completely visible. When editing a program, this window moves to the top of the "stack" of windows, displays the lines of your program and allows you to correct errors. Details on editing your programs are included in the *If You Make A Mistake* section of this chapter.

The functions of the Edit window and the Command window overlap somewhat. Entering new program lines and using commands like RUN and LIST when either the Edit window or the Command window is activated.

Activating Windows

Let's take a closer look at the windows themselves. Figure 10.1 shows the BASIC desktop. When you load ST BASIC the top of the Command window is shaded. That shaded area means this window is activated. When a window is activated you can close it, change its size, or move it to another area on the desktop.

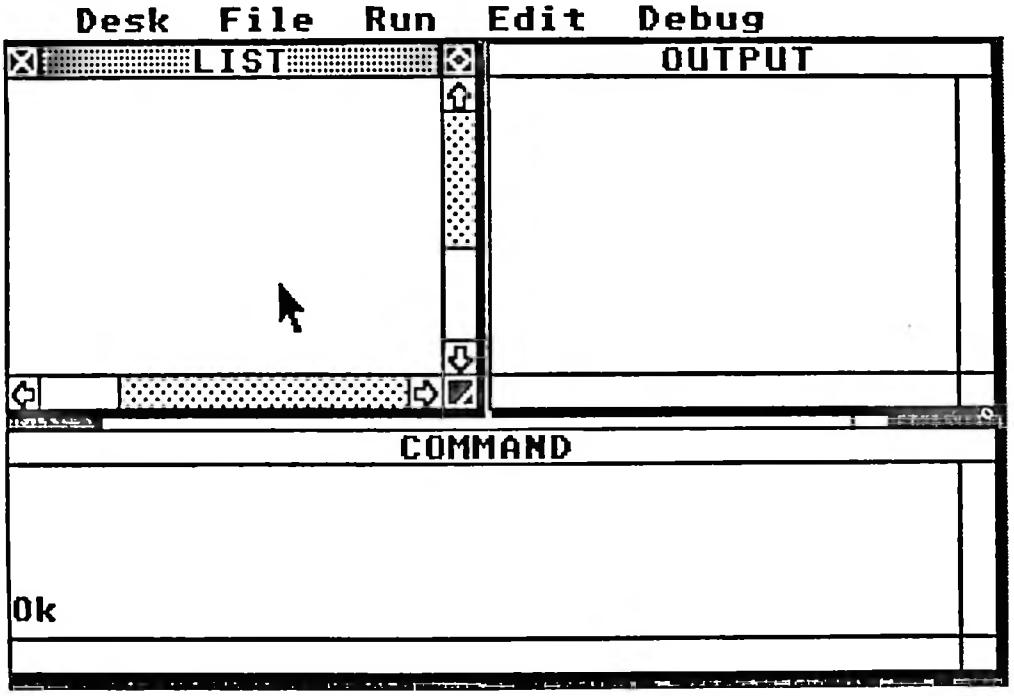


Figure 10.2-BASIC screen with List Window Activated.

To activate the List window move the pointer into the List window and click the left mouse button once. Now the shaded area should appear across the top of the List window. This same procedure works to activate any of the windows. Just click the left mouse button with the pointer positioned in the window you wish to activate.

Manipulating Windows

In an earlier chapter you learned how to manipulate the ST desktop windows by using the Size box in the lower right corner, the Full box in the upper right hand corner and the vertical and horizontal Scroll boxes and arrows. These same procedures can be used to modify the windows on the ST BASIC desktop.

Viewing Material in Windows

Sometimes the lines displayed on the screen will be too long to fit within the window. For instance, the output window is only 35 characters wide and some

of the lines in your program will output more than 35 characters. You can view the entire line by changing the size of the window or the width of the line.

Opening and Closing Windows with Commands

Each of the four windows on the BASIC desktop has a number. This table shows the number for each of the windows:

Window	Number
Edit	0
List	1
Output	2
Command	3

These window numbers are used in the OPENW and CLOSEW commands. To use the OPENW command to open the List window you would type

```
OPENW 1
```

or you would type

```
CLOSEW 3
```

to close the Command window. When you type these or any other commands, they will appear in the Command window. The OPENW command is the only way to open a window on the BASIC desktop once it has been closed.

You can also close the windows by putting the mouse pointer on the Close box in the upper left hand corner of the window and clicking the left mouse button. But, be careful. We found out by accident that closing all four windows on the BASIC desktop is a mistake. If you happen to do that, the only way to reactivate the windows is to turn the computer off, turn it back on, and load BASIC again.

One more command is useful when working with BASIC's windows. The CLEARW command is used to clear a window.

DIRECTING THE ST IN BASIC

When you want the computer to do something, you use the keywords the computer understands to create statements. Statements correspond to sentences in English. One of the most common tasks you tell a computer to do is to display something on the video monitor. The simplest way to tell the ST to do that is with a PRINT statement. As mentioned above, a statement in BASIC is like a sentence in English. Load ST BASIC following the directions given previously, and when the OK prompt appears on the screen, type in the following statement and press the RETURN key. (Replace the name we used with yours in the statement.)

```
PRINT "MY NAME IS GWEN BRUNSON."
```

The statement should appear in the Command window as you type it. When you press the RETURN key, you will see the sentence:

```
MY NAME IS GWEN BRUNSON.
```

displayed in the Output window. (Note: If you make a typing error, see the section of this chapter entitled *If You Make a Mistake.*)

PRINT is a keyword that instructs your computer to display what follows in quotation marks on the screen. You can also tell the ST to perform calculations and display the answers by using PRINT. Try this example:

```
PRINT 20 + 40
```

When you type this and press the RETURN key, all you see displayed in the Output window is the result of the computation, the number 60. What happens if you type the numbers inside quotation marks? Let's try it and see. Type:

```
PRINT "20 + 40"
```

Now when you press RETURN, 20 + 40 is displayed. As you can see, the computer will display anything it reads inside quotation marks when you use the keyword PRINT.

Immediate and Program Modes

The lines you have typed in are forgotten once the result is displayed. You can continue to type in more statements, but the computer will not store them in memory. This is called immediate mode. You type in one statement or direction and the computer carries out--executes--the command as soon as you press the RETURN key.

Longer, more complex directions are possible only if you write them in program mode. The lines you enter in program mode are similar to the lines you have been entering in immediate mode, but to get the computer to store your statements in memory and then execute all of them, you must use line numbers.

Your First Program

Let's try typing in a short program. Notice the following program lines are printed in all capital letters. The ST doesn't care whether you use upper case or lower case letters when you enter commands. Always press the RETURN key after typing a line to tell the computer you are finished with that line. The ST will move the cursor down one line. The cursor always indicates the position on the screen where the next character typed will be displayed.

Type these statements:

```
10 PRINT "I CAN DO MATH PROBLEMS."
20 PRINT "WHAT IS 20 + 40?"
30 PRINT 20 + 40
```

IF YOU MAKE A MISTAKE--ST BASIC EDITING

If you are like us, you tend to make typing errors now and then when entering information from the keyboard. You probably will make some as you type in this program. If you do, there are several ways to correct the mistakes. In this section you will learn how to correct those mistakes in immediate mode and you will learn about some special help ST BASIC offers in the Edit mode.

Before Pressing RETURN

Sometimes you will notice an error as soon as you type it or at least before you press the RETURN key when you finish the line. If that happens, you can press the BACKSPACE or DELETE keys to erase characters back to the error in the line. Then, all you have to do is type the rest of the line again correctly.

After Pressing RETURN

There will be times when you do not notice a mistake until after you press the RETURN key to tell the computer you are finished with that line. When you make a mistake entering a line and then press RETURN the ST will display the error message "Something is wrong" in the Command window. A little carat symbol (^) appears pointing to the section in the line where the error occurred.

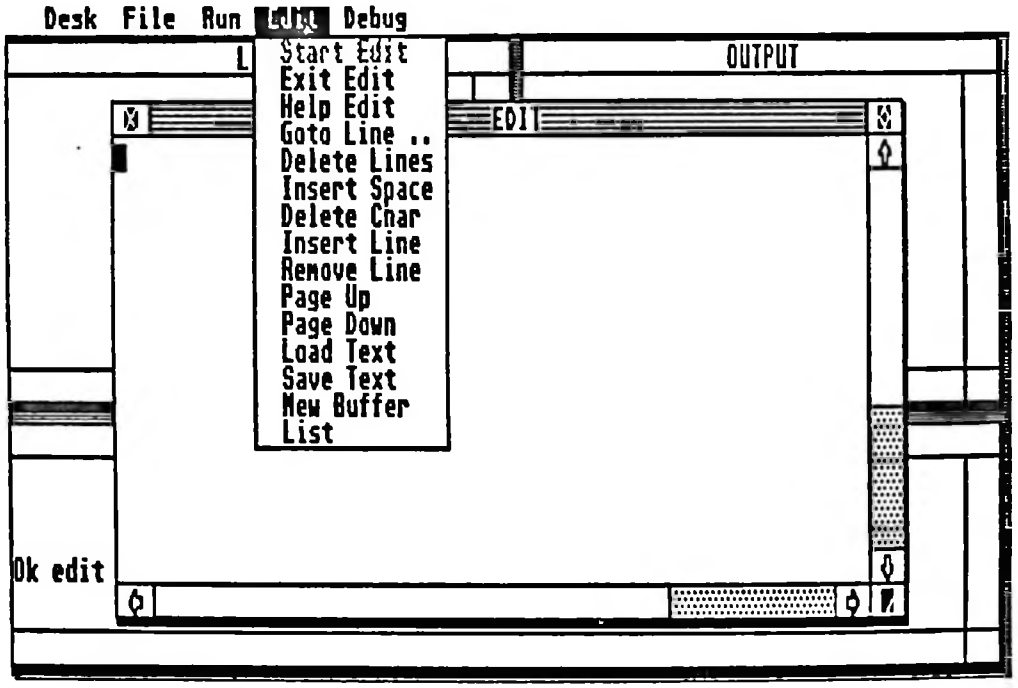


Figure 10.3-The BASIC Edit Menu.

The easiest way to correct this kind of error is to retype the line. Suppose you typed AS instead of IS when you typed line 20 in this short example program. If you retype the line correctly and press RETURN, the new version of line 20 will replace the old version in the computer's memory. The old version will still be displayed on the screen but the ST accepts the change. It always remembers the last version of any line you enter.

EDIT Mode

Retyping entire lines when you make an error can be tiresome and time-consuming. ST BASIC has a special Edit Mode which offers some alternatives to correcting errors in your programs. You will notice that Edit is one of the options on the menu bar at the top of the BASIC desktop. Figure 10.3 shows the Edit menu.

You can see the options on this menu by using the mouse to "pull the menu down" or by typing ED or EDIT from the keyboard.

To begin editing you must first click on the Start Edit option on this menu. When you choose this option, the Edit window is activated, expands and moves on "top" of the other windows on the desktop. The lines of the program in the computer's memory are displayed in the Edit window. Now, you can use the arrow keys to move the cursor to any line in the program. Let's try editing a program line. Type:

```
40 PRONT "THIS IS EDITING PRACTICE."
```

just as it appears, mistake and all. Then you press the RETURN key at the end of the line. Now move the mouse pointer to the Edit menu and select the Start Edit option. You should see the lines of your program in the Edit window as shown in Figure 10.4. Now, use the down arrow key to move the cursor to line 40 of your program, then use the right arrow key to move the cursor to the O in PRONT. All you have to do is type I and it will replace the O.

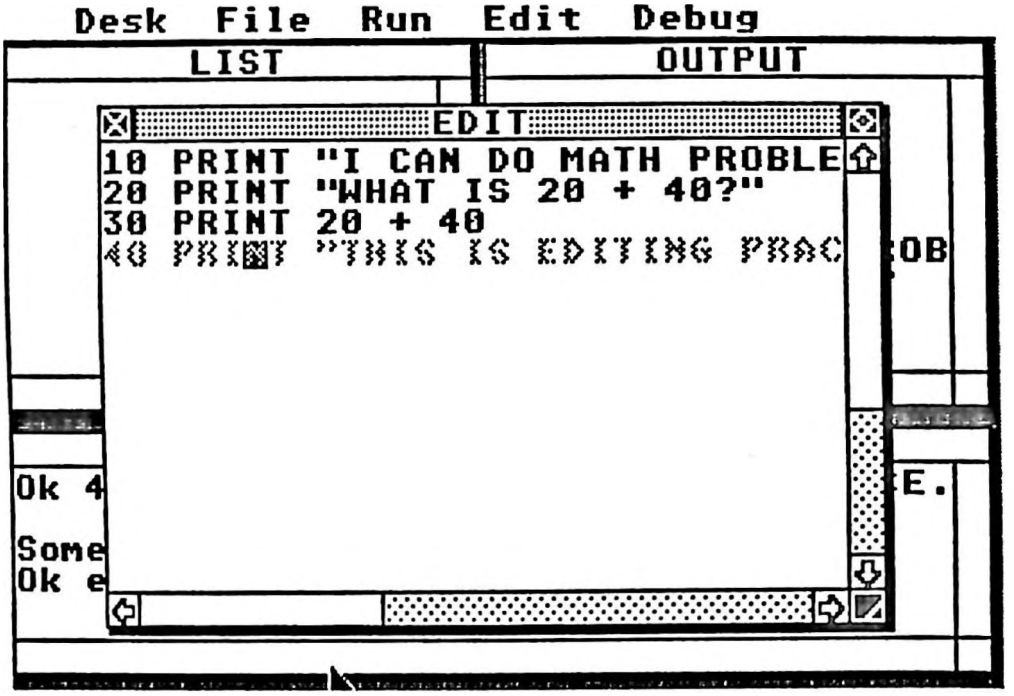


Figure 10.4-Editing a program listing in the Edit Window.

There is one thing we don't like about this method of editing. In the Edit mode when you move the cursor to a line of your program and execute one of the editing options, the line you are editing changes color. The color changes to a lighter shade and the letters themselves change to some sort of Italic-looking characters. The color and type font combination make the lines you edit difficult to read.

Function Keys in Edit Mode

The time you spend moving the mouse back and forth to the Edit menu can be eliminated if you feel more comfortable using one-key commands. Move your

mouse pointer to the Edit Menu and click on the Help Edit option. A dialog box appears in the middle of the screen showing the same editing commands found on the Edit menu itself. But this dialog box shows you which of the function keys--located across the top of the keyboard--can be used to execute those commands. Figure 10.5 shows the Help Edit dialog box.

For instance, pressing the F2 key deletes the character at the cursor just like choosing the Delete Character option on the Edit menu does. (The characters in the lines you edit still change color and shape when you use the function keys.) If you do a lot of programming, you may choose to use the function keys when you are editing. One-key commands save a lot of time. And, when you use them frequently you will soon remember which function key does what without looking at the dialog box.

You can add lines to, RUN, or LIST your program while you are in the Edit mode. But, if you want to discontinue the Edit mode and return to using the Command window, you must select the Exit Edit option. This option is found on the Edit menu and in the Help Edit dialog box.

Now let's get back to that first program you have entered. First, type 40 and press the RETURN key to erase the editing practice line from your program. The three remaining statements with their line numbers do make up a short program. A program does not have to be complicated, it is simply a set of statements that tell the computer how to do something.

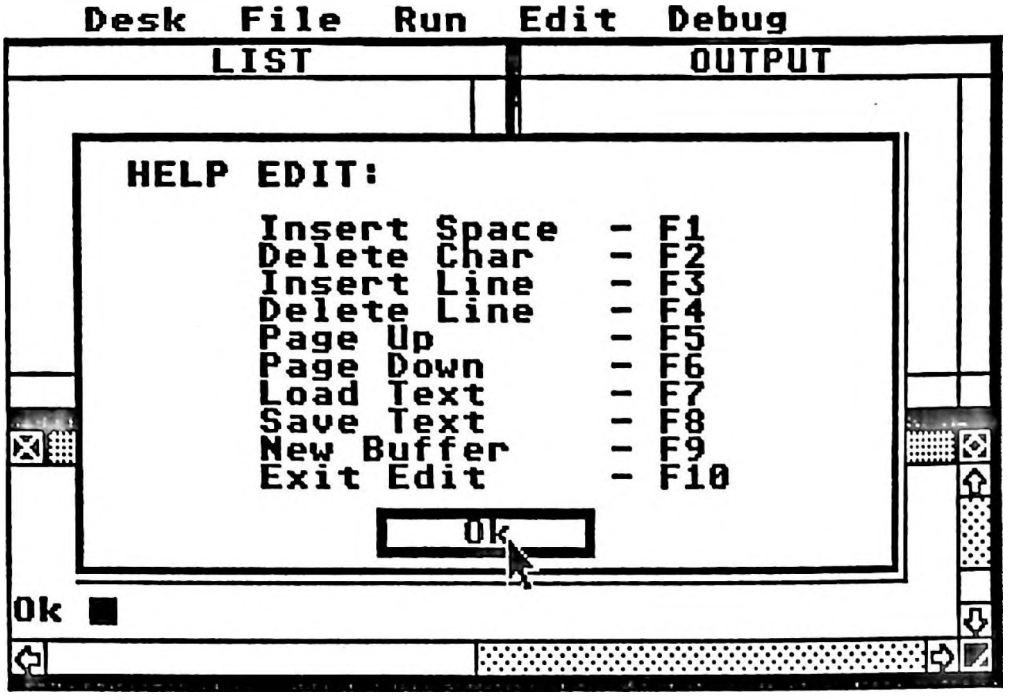


Figure 10.5-The HELP EDIT dialog box showing uses of function keys.

The first part of each statement in a program is a line number. The first line in a program can be 1. However, most programs will usually begin with line number 5 or 10. Additional line numbers may be numbered consecutively. But if you begin with line number 5 and number the next lines 6,7, 8 and so on you may run into problems. If you needed to add a line between lines 6 and 7 later you would have to renumber the whole program since line numbers cannot have deci-

mals. Most programmers find it easier to use line numbers which increase by 5 or 10 (e.g. 5, 10, 15, 20, or 10, 20, 30, 40). That leaves room for any additional lines which might need to be added to the program later.

RUNning Programs

To see the result of a program, you must tell the computer to RUN it. There are two ways to do this.

1. Type RUN and press RETURN.

or

2. Select the RUN menu from the menu bar at the top of the screen and click on the first menu option, RUN.

Either method causes the computer to begin with the lowest line number, execute the instructions in that line, and move on to the next higher line number. It will go through each statement until they are all executed. When you RUN the program you entered, your screen should see look similar to Figure 10.6

Line 10 is a PRINT statement like the ones you used in immediate mode. Anything inside quotation marks is displayed on the screen when the computer executes that statement. In this program the computer executes line 10, then goes to line 20. It executes this line by displaying WHAT IS 20 + 40? Then, when it reaches line 30, it does the addition and displays the answer.

You can use what you have learned in this section to write your own simple programs. When keywords are used to tell the computer to do something immediately, they are called commands. When they are included in program lines, they are usually called statements. BASIC programs must contain statements with line numbers. The statements must use keywords and follow the punctuation rules BASIC understands. Most of the keywords, like PRINT, can be used as commands as well as statements. In other words, you can use them in either immediate mode or program mode.

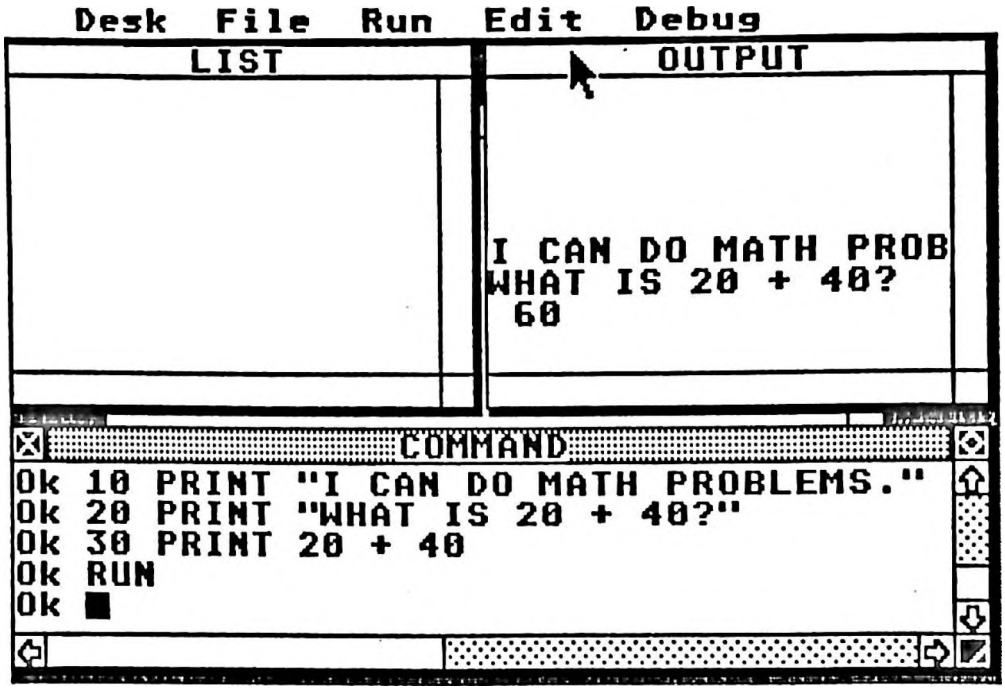


Figure 10.6-The result of program appears in the Output Window.

MORE COMMAND WORDS

As you write programs or use programs other people have written, you will need to communicate with the computer to tell it exactly what you want to do with the program. You have already learned how to LOAD and SAVE programs in Chapter 4. You have used the RETURN key and have RUN programs. The following is a list of these and some of the other more frequently used BASIC keywords you will use when working with programs.

RUN

You have already used this keyword several times. It causes the computer to execute the program in the computer's memory starting with the instructions in the line with the lowest line number. You can also use RUN to start executing a program at some place other than the beginning. If you type RUN 100, the computer will skip all lines before 100 and begin executing the program at that line and continue execution to the end of the program. You can use this command to begin at any line number that appears in the program.

In immediate mode you can type

```
RUN filename
```

to tell the computer to load and execute the file with the name given.

NEW

The NEW command erases any program in the computer's memory. This command is used when you want to start a new program. If a program is already in the computer's memory and you begin working on a new one without using the NEW command, the programs may overlap and cause some confusing results.

There is no way to recover programs you erase with the NEW command if they are not saved on a disk. Therefore, don't type NEW before saving the program to a file on disk if you want to keep a permanent copy of the program.

SAVE and LOAD

These keywords, covered in Chapter 4, cause the computer to store a program on, or read a program from, the diskette. Remember these commands are available on the File menu on the menu bar at the top of the BASIC desktop. You can execute these commands by using the mouse to click on the words on that menu. Or, you can type in the commands in immediate mode. Typing

```
SAVE filename
```

will accomplish the same task as choosing the Save As command on the File Menu, giving the program a name and saving it to the disk. And, typing

```
LOAD Filename
```

in immediate mode does the same thing as choosing the Load option on the File Menu, loading the program with the specified name from the disk.

END

This word signals the computer that it has reached the last program line to be executed. The computer will stop executing instructions any time it encounters this keyword. Many programmers always put END in the last line of their programs. You may use this technique but it is not necessary. The computer will stop when there are no more program lines even if END is not in the last line.

PROGRAM LINE NUMBERING COMMANDS

ST BASIC has several command words which make entering and deleting program lines easier. You may find the following commands helpful as you write your own programs.

AUTO

If you use the AUTO command, the computer will automatically number the program lines. In immediate mode type AUTO and press RETURN. Line number 10 should appear at the beginning of the next line in your Command window. When you use the AUTO command before you begin entering a program, the computer will automatically begin the program with line number 10 and will increment the following line numbers by 10 (i.e., 20,30,40...). You may also specify a starting line number and a different increment in the command line.

To stop the automatic line numbering, you simply type Control/G. (Control/G means you should hold down the Control key on the left-hand side of the keyboard and press the G key.)

DELETE

Sometimes it is necessary to delete a line from your program. To delete one line of a BASIC program simply type the line number and press RETURN. The old line will still be on the screen but if you LIST the program you will see that the new line has replaced the old one in the computer's memory. If you need to delete a series of lines, use the DELETE command. You can choose the Delete Line option from the Edit menu. If you choose this command, a dialog box appears asking you to enter the beginning and ending numbers of the lines to be deleted. You can also enter the DELETE command from the keyboard.

RENUM

Once you have deleted lines, you may want to renumber the remaining lines of your program. ST BASIC includes a keyword, RENUM, that makes this task simple. If you type RENUM, the computer will automatically renumber the program. The first line will be numbered 10 and each additional line will be numbered in increments of 10. To renumber only a part of the lines of your program, the RENUM command must include more information.

DEBUGGING PROGRAMS

Sometimes a program you have written or typed from another source doesn't execute properly when you RUN it. The problem may be caused by a typing error, incorrect use of a command, or any number of other possible errors. ST BASIC has some helpful commands that make finding the errors, called bugs, in programs much easier. In fact, the debugging commands are available on the Debug

Menu. You can use these commands by choosing them from this menu or by typing them in from the keyboard in the immediate mode.

TRON and TRACE

Your program may execute without generating an error message, but it may not do what it is supposed to do. In such cases, it is sometimes helpful to follow the flow of the program as it executes each line. You can do this by using the TRON or TRACE commands. Type this short program:

```
10 PRINT "WHAT HAPPENS IN LINE 10?"
20 PRINT "THIS IS LINE 20."
30 PRINT "TRACE WORKS LIKE THIS."
```

If you type TRACE in immediate mode. or click on the TRACE option on the Debug menu, a dialog box will appear on the screen. You may choose to have the computer trace all the lines in your program as they execute or, if you only want to trace a segment of the program, you may choose to limit the trace to part of the program by entering the line numbers of the segment you want to trace. This time we want to TRACE all the program lines, so click on the box which says ALL in the dialog box. Once you have made that choice, click on OK in the dialog box. Try RUNning your program. The program lines will appear in the Command window as each instruction is executed as shown in Figure 10.7

You can turn off the TRACE by clicking on UNTRACE on the Debug menu or by typing UNTRACE in the immediate mode. Notice the TRACE command displays the entire line as it executes. If you try using the TRON command, you will find there is only one difference between the TRACE and TRON commands. The TRON command works the same way but it displays only the line number (not the entire line) as each line executes.

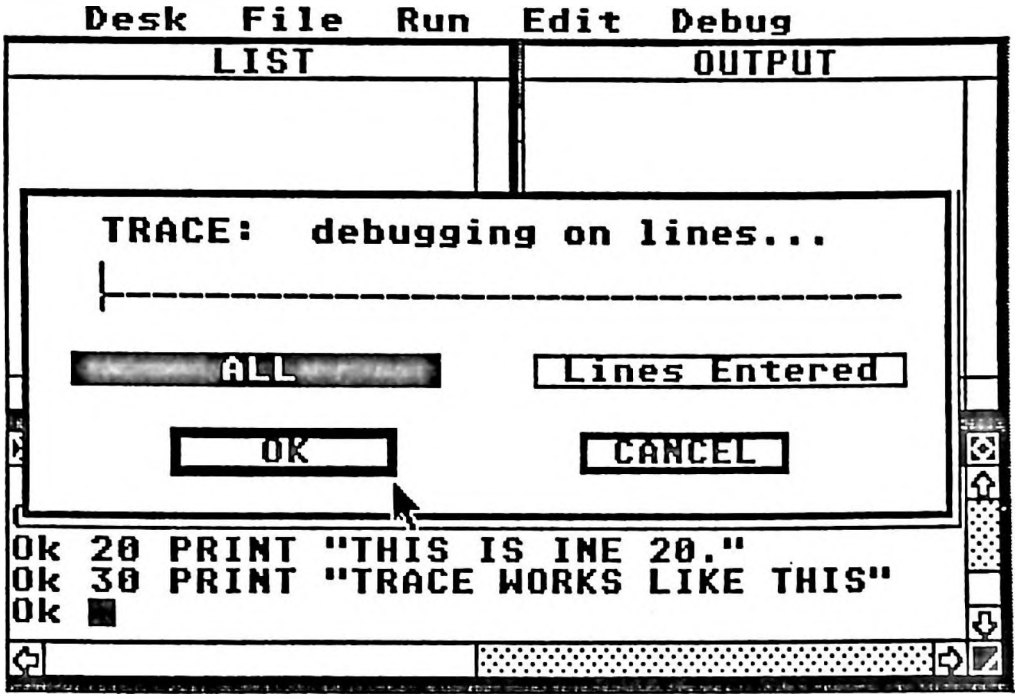


Figure 10.7-The TRACE dialog box.

These commands can be helpful in finding problems within a program. However, the windows on the ST BASIC desktop can be somewhat distracting. When you are using the TRACE or TRON commands, the program lines being executed are displayed in the Command window. The results of that execution are displayed in the Output window.

We had changed the Output window to full screen size before trying the TRACE command. When the program started executing, the Command window appeared to display each program line, quickly disappeared to show the result of executing the line in the full screen Output window, then just as quickly reappeared to display the next line. The Command and Output windows almost seemed to "flash" on and off the screen not showing either window long enough to read the display. If you are going to use the TRACE or TRON commands, be sure the window sizes allow for displaying the program lines and their output on the screen at the same time.

SCREEN DESIGN COMMANDS AND RULES

The PRINT statement you have used in the sample programs displays characters on the screen. Unless you tell the computer to do otherwise the information included in a series of PRINT statements will be displayed on consecutive screen lines. That is usually the way it is done, but BASIC has several keywords and punctuation rules that control the screen display. They allow you to vary the placement of information in your program.

WIDTH

ST BASIC allows you to set the width of the line which is output to the screen. The normal, or default, value for the width of the line displayed is 72 characters. However, the WIDTH command can be used in program mode to set the number of characters displayed.

Semicolons and commas

Using a semicolon or a comma in a PRINT statement causes the characters following it to be printed on the same line as the information contained in the statement. Type the following lines:

```
10 LET X = 45
20 PRINT X
30 PRINT X+3;X+5
40 PRINT X+3,X+5
50 PRINT "X= ";X;X+3;X+5
60 PRINT "X= ",X,X+3,X+5
```

If you will type the FULLW 2 command before you RUN the program so that you can see the full line width. When you RUN the program, you should see the following lines

```
45
48  50
48            50
X= 45  48  50
X=            45            48            50
```

in the Output window. (If you did not change the size of the Output window, the 50 in the output from line 60 will not be visible since the window does not display that many characters. Scrolling the Output window display to the left with the Scroll Box or Scroll Arrow at the bottom of the window will show the rest of the line.)

The results show you how BASIC responds when you use commas and semicolons. Line 10 illustrates the LET command. In this program it sets the value of the variable X to 45. (You will learn more about LET and variables in the next section.) Line 20 tells the computer to print the value of X (45).

Usually the computer will skip to the beginning of the next line once it executes a PRINT statement. The semicolon and comma can be used to tell the computer to print information on the same line instead of moving down to a new line for the next PRINT statement. The semicolon in line 30 tell the computer to continue printing material on the same line. The commas in line 40 tell the computer to skip to the next print zone before printing the material after the comma.

Characters inside quotation marks in a PRINT statement are called a literal string. The computer literally prints the string of characters enclosed in quotation marks. Lines 60 and 70 both print the literal string X=. The display resulting from execution of these lines is different because of the punctuation used. The computer does not react to the semicolon after the literal string in line 50 as we might expect.

This is somewhat confusing, but when it comes to the semicolon after a literal string, the computer skips only one space before printing the material after the semicolon. Then it skips two spaces when it encounters the other semicolons in the line. Line 60 executes the commas the same way as line 40, telling the computer to skip spaces before printing the next information.

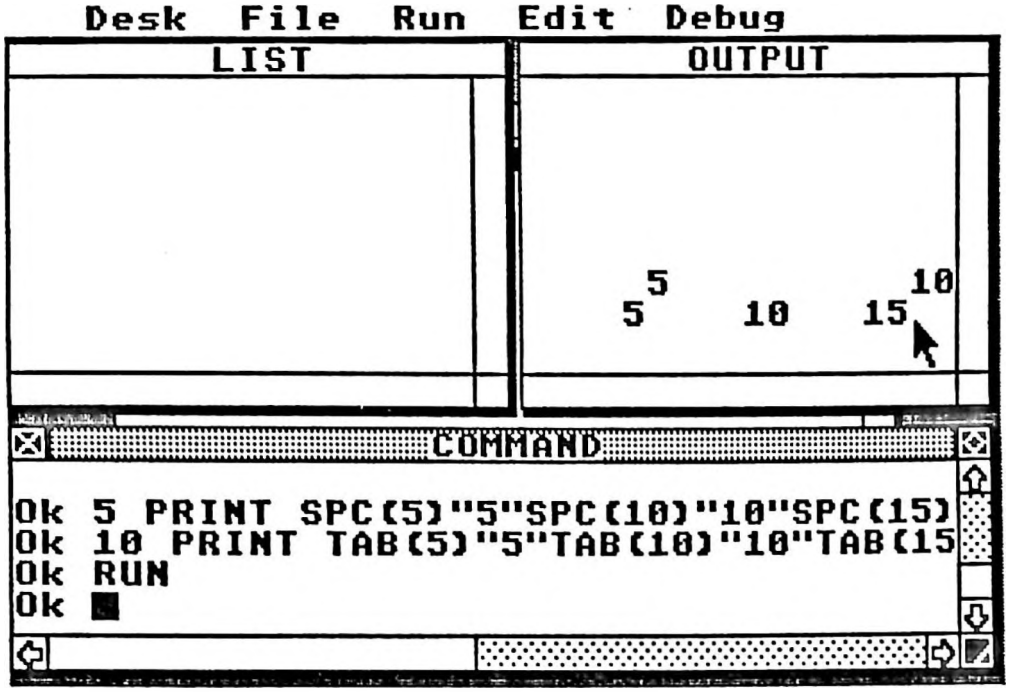


Figure 10.8-Output examples of SPC and TAB.

SPC and TAB

The SPC and TAB commands offer even greater control over where material is displayed on the line. Use the NEW command to clear the computer's memory. Clear the Output window with the CLEARW command and the computer's memory by typing NEW. Then, type this short program.

```
5 PRINT SPC(5)"5"SPC(10)"10"SPC(15)"15"
10 PRINT TAB(5)"5"TAB(10)"10"TAB(15)"15"
```

Figure 10.8 shows the result of this program. Output from the two program lines is different because the SPC and TAB commands work differently.

Line 5 causes the computer to skip over 5 spaces on the line, print 5, then skip 10 more spaces, print 10, then skip 15 more spaces and print 15. The line includes a total of 35 characters counting the blank spaces. The SPC command tells the computer to skip a specified number of spaces *from the cursor position.*

The TAB command tells the computer to skip a specified number of spaces from the beginning of the line. Line 10 causes the computer to print the number five spaces from the beginning of the line, the number 10 ten spaces from the beginning of the line, and the number 15 fifteen spaces from the beginning of the line. The display of the output from line 10 covers only 17 spaces on the line.

GETTING INFORMATION INTO THE COMPUTER

Some of the information used in a program may change within the program or may be different each time the program is executed. This type of data is often stored under names called variables until it is used in the program. Programs can assign values to variables in two ways. In this section you will learn about the types of variables and using LET and INPUT to assign values to them.

Using Variables and LET

The instruction PRINT 50 tells the computer to display the number 50 on the screen. The number 50 is a constant because its value is always the same. If you ran a program with this line in it, the result would always be the same since the value of the number does not change. This line would print the number 50 every time.

However, some values may change within the program or may be different each time the program is run. If the program asks "HOW OLD ARE YOU?", the answer could vary from user to user. Since the response may vary, the computer is usually programmed to store the response as a variable.

LET

You have used the LET keyword in one of the sample programs. LET tells the computer what value to associate with a particular variable name. It can be used to assign values to either numeric or string variables. The format for LET is:

```
LET variable name = value
```

For example, LET A=20 tells the computer to remember the variable A is equal to 20. The computer will store that value for A until it is given another value for A.

In this example, A is a numeric variable since its value is a number which can be used in computations.

This statement

```
LET A$ = "UNITED STATES"
```

assigns a value to a string variable. The computer must have some way of distinguishing between numeric and string variables. The $ is used to identify a string variable. The $ after the A in the example tells the computer to expect words or letters as the value of this variable. Numeric variable names do not include the $, so the computer will assume the variable is numeric rather than string if you leave off this important identifier. In LET statements, you must enclose the values of string variables in quotation marks--LET B$ = "LIKE THIS", instead of LET B$ = LIKE THIS.

ST BASIC reads variable names up to 31 characters long. Names like TEST1, EXP, and G6 are acceptable numeric variable names. COMPUTER$, DATE1$, and G$ are examples of acceptable string variable names. There is really only one restriction to consider when deciding what names to give variables. BASIC keywords such as PRINT, LET, and RUN cannot be used as variable names. The instruction PRINT$ = "YES" will not work because PRINT is a keyword.

In the example, A$ is now assigned the value UNITED STATES. If the computer executed the instruction PRINT A$, UNITED STATES would be displayed on the screen.

The ST will not accept LET B = "UNITED STATES" because there is a mismatch between the type of variable specified (numeric) and the type of value assigned (string). If you type LET B = "UNITED STATES" and press the RETURN key, the computer will respond with a "Types of values do not match" error message.

Values assigned to a variable can be changed within a program. You can assign one value to A (or A$) early in a program--10 LET A=50--and then change the value later in the program--60 LET A=25. The value of A will always be the last value assigned (25 in this example).

The LET statement can also use the value of one variable to assign a value to another variable. Look at this example:

```
10 LET A = 25
20 LET B = A
30 LET C = A + B
40 PRINT A,B,C
```

When you RUN this short program, line 20 assigns a value to the variable B by making it equal to the variable A. Line 30 illustrates the use of an arithmetic expression--like A + B in this example--in a LET statement to assign a value to another variable.

The computer must have a value for any variable used on the right side of the equal sign in a LET statement. The variable B was assigned a value (25) in line 20 of this program. Line 40 establishes a value (50) for C. Then line 40 displays the values of the three variables on one line.

There is one last important thing to remember about using LET. You really don't have to use it! These two lines LET A = 25 and A = 25 mean the same thing. In the second example the keyword LET is understood. The computer will execute the line even though the word LET is not included in the statement.

INPUT

Sometimes you may want the person using your program to enter responses to questions. In order to allow the user to enter answers you should use the INPUT command. The computer will stop execution of the program and wait for someone to type in a response and press RETURN when it comes to an INPUT statement. If INPUT is followed by a numeric variable name--INPUT A or INPUT SCORE--the computer expects the user to type a number and press RETURN. If INPUT is followed by a string variable name--INPUT GR$ or INPUT PURCHASE$--the computer expects a letter or word in response. Enter this program:

```
10 PRINT "WHAT IS YOUR NAME?"
20 INPUT N$
30 PRINT "HELLO, ";Ns;" HOW ARE YOU TODAY?"
40 INPUT "HOW OLD ARE YOU?"
50 PRINT "AGE: ";A
```

RUN this program. You will see WHAT IS YOUR NAME? on the first line of the display. A ? will appear on the next line to signal the computer is waiting for a response. When you type your name and press RETURN, the computer will set the value of N$ to the name you typed.

Line 30 shows you one way to display the value of a variable in a PRINT statement. Remember the PRINT statement will print any characters within quotation marks. Line 30 displays HELLO, then the value of N$ next to that, and, finally, HOW ARE YOU TODAY?

The computer goes on to execute the remaining lines once you have entered your response. Line 40 is another example of an INPUT statement. This line is actually a PRINT statement and an INPUT statement combined. When the INPUT statement is in this format, anything inside the quotation marks is displayed on the screen just as though it was in a PRINT statement. But, since the computer encounters the INPUT keyword in the line, program execution stops until a response is entered.

You have now learned some of the BASIC programming techniques that allow you to create your own programs or to understand other programs written for the ST. This chapter does not cover all ST BASIC commands. The next chapter will introduce you to other BASIC commands and help you learn to create sound and graphics with BASIC.

11

More Basic, Sound and Graphics

In this chapter we will discuss more BASIC programming. You will learn how to control the flow of your program and how to use keywords to create graphics and sound.

COMPARING VARIABLES WITH IF...THEN

IF...THEN statements cause the computer to make a decision about what it should do next. A simple IF...THEN statement is shown below:

```
IF A = 8 THEN PRINT A
```

The first part of this statement is a condition. When the computer comes to this line in a program it checks the value of A stored in memory. IF that value is 8, the condition is met, meaning it is true. Then the computer will execute the rest of the statement and PRINT A. The computer will perform the action after THEN only IF the condition is true. IF the condition is not true--in the example, if A is not equal to 8--THEN the computer moves on to the next program line.

This short program uses IF...THEN to set up a counter:

```
10 N = 1
20 PRINT N
30 N = N + 1
40 IF N < 10 THEN GOTO 20
50 PRINT "STOPS AT NINE"
```

Lines 10 and 20 name a variable (N), assign a value to that variable (1), and print the value. Line 30 is a counter. Each time the computer comes to this line the value of N increases by 1. The IF...THEN statement in line 40 tells the computer to check the value of N and see if that value is less than 10. If this condition is true and N is less than 10, the computer executes the instruction after THEN and jumps back to line 20. If the condition is not true (A is not less than 10), the next line of the program is executed.

Arithmetic Conditioners. This example uses the equal (=) and the less than (<) conditioners. You can use any of these arithmetic conditioners:

= equal to
<> not equal (more or less than)
< less than
> more than
<= less than or equal
>= more than or equal

This program illustrates how some of these conditioners work.

```
10 PRINT "LET ME DO SOME MULTIPLICATION FOR YOU.
15 INPUT "TYPE A NUMBER BETWEEN 1 AND 20.";N1
20 IF N1<1 OR N1>20 THEN PRINT "TRY AGAIN":GOTO 15
25 INPUT "NOW TYPE ANOTHER NUMBER.";N2
30 PRINT "I'LL DO THE MULTIPLICATION FOR YOU."
35 PRINT "THE ANSWER IS "; N1*N2
```

When this program is executed, you are asked to type a number between 1 and 20 so the computer can do some multiplication for you. When executing line 20 the computer checks to be sure the number you typed is between 1 and 20. If your number does not fit that condition, you will be given another chance to type a number. If your number does fit that condition, you are asked to type another number. Then the computer shows off its computation speed by multiplying the two numbers you entered and showing you the answer.

Logical Conditioners. Notice that line 20 uses a "logical conditioner" (OR) in the IF...THEN statement. When OR occurs in a statement like this, the computer will perform the activity after THEN if either condition is true. In this example, IF the number you type is less than 1 the computer will print the prompt TRY AGAIN. OR if your number is greater than 20, you will see the same prompt.

Another logical conditioner, AND, can be used to instruct the computer to perform the activity after THEN only if all conditions in the statement are true. We might change line 20 to look like this:

```
20 IF N1 > 1 AND N1 < 20 THEN PRINT "GOOD JOB!"
```

The program would work in exactly the same way. Now the computer checks to see if the number you type is both greater than 1 AND less than 20. If both conditions are met, then the program executes the PRINT statement and displays GOOD JOB!

Most of the examples have used a PRINT instruction after THEN, but you can use any other BASIC keyword. You could tell the computer to GOTO another program line or to perform a math operation (e.g., IF...THEN X=X+1).

FOR...NEXT LOOPS

Another way to control the flow of a BASIC program is with FOR...NEXT loops. A loop causes the computer to execute a specific part of the program more than once. In the last section we used IF...THEN to set up a counter. The following program illustrates a different way to set up counters, using a FOR...NEXT loop.

```
10 FOR I = 1 TO 10
20 PRINT
30 NEXT I
```

```
40 END
```

Run this program and the numbers from 1 to 10 will be displayed on the screen.

Let's examine the FOR...NEXT loop closely. The FOR instruction tells the computer how many times to execute the lines inside the loop. The NEXT in line 30 is the lower boundary of the loop. When it comes to NEXT, the computer will return to the line where FOR occurs and go through the loop again.

The FOR line begins the loop and establishes the variable name of the loop. This variable is called the control variable. In this example, I is the control variable, but you can use any acceptable variable name. The expression on the right side of the equal sign tells the computer where to begin counting and how many times to execute the loop. Line 10 tells the computer to set the variable I equal to 1. Then line 20 is executed and the computer displays the value of I on the screen. When the computer reaches line 30, it checks the value of I and compares that value to the final value in the FOR statement. In this example the final value is 10.

If the value of the variable is less than the final value--for example, if I<10--the computer returns to line 10 and goes through the loop again executing each program line between FOR and NEXT again. Each time the computer comes to line 30, the value of I increases by 1, counting the number of times the loop is executed. When I is equal to 10, the computer continues through the loop. When it reaches the NEXT statement in line 30 it checks the value of I. Since I is equal to 10, the program goes on to the next program line and stops looping.

Every FOR statement must have a matching NEXT statement or the loop will not execute. In the example, the control variable increases by one each time the loop is executed. There may be times when you do not want it to increase by 1. You can control the amount of variable increase by using STEP.

GOSUB...RETURN

In addition to the GOTO statement, BASIC provides another method of executing a section of your program more than once. GOSUB, like GOTO, tells the computer to jump to another section of the program. However, GOSUB works differently.

Some programs must execute the same instructions several times in different parts of the program. Instead of typing the same lines into your program in several different places, the use of subroutines allows you to type the frequently-used lines only once.

This program contains an example of a subroutine:

```
15 PRINT "GRAMMAR ACTIVITY"
20 PRINT "TYPE THE PLURAL OF TURTLE."
25 INPUT P1$
30 IF P1$ = "TURTLES" THEN GOSUB 1000
35 PRINT "SPELL THE PLURAL OF MOUSE."
40 INPUT P2$
45 IF P2$ = "MICE" THEN GOSUB 1000
50 PRINT "THAT'S ALL FOR NOW."
55 END
1000 PRINT "THAT'S CORRECT!"
1005 PRINT "TRY ANOTHER ONE."
1010 RETURN
```

This program presents two questions which ask you to enter the plural form of a noun. Lines 20 and 35 display the questions. Lines 25 and 40 allow you to enter a response. Lines 30 and 45 check to see if your response is correct and, if it is, display a reward statement.

When the computer comes to lines 30 and 45, the GOSUB instruction causes the program to jump to line 1000 and begin executing the program lines in order until it reaches the RETURN keyword. Then the program returns to the line following the GOSUB statement and continues executing from that line. We used 1000 in this program but the subroutine can start at any line number. Some programs may need several different subroutines. Each subroutine must have a matching RETURN statement to signal the end of the subroutine or the program will not execute properly.

The difference between the GOTO statements used before and the GOSUB statement is the RETURN. GOTO statements cause the program to jump to another program line and continue program execution from that point. GOSUB statements cause the program to jump to another program line and execute the instructions until it reaches a RETURN statement. Then the program will return to the line following the GOSUB statement that sent it to the subroutine.

We could expand the example program to ask 50 questions about noun plurals and use the same subroutine to reward a correct answer each time. But, we would still need to type the lines in the subroutine one time instead of 50.

Subscripted Variables

Another major BASIC concept we will cover is subscripted variables. As you have learned from the examples in the last two chapters, variables always have names and values. You have worked with both string and numeric variables.

Several programs have used more than one variable. The names in a program may have looked similar to us (e.g., N1, N2) but the similar names were of no importance to the computer. No special instructions exist to tell the computer to perform some task with all the N variables, for instance. Suppose you wanted to set up a program to print the names and prices of your software inventory. You want the program to print out in a two-column list. You would need some way to tell the computer to make the two lists. BASIC provides a way of communicating this to the computer, subscripted variables.

The variables we have used in our example programs have ordinary names made up of letter or letters and numbers. Subscripted variable names look a little different. N(1), NAMES$(4), and DATE(2) are examples of subscripted variable names. The variable N1 is an ordinary numeric variable. The variable N(1) is a subscripted variable; N is the variable name and 1 is the subscript.

Think of the data in the computer as seats in an auditorium. The variable S4 might identify any one seat in the auditorium. The variable S(4), however, designates a specific seat in a specific row. The seats beside it are S(5) and S(3). The first seat on that row would be S(1) and one farther down the row might be S(8). If this row of seats were the data in a computer they would be called a list or one-dimensional array since they are a group arranged in one long line.

Using subscripted variables helps you organize the data in the computer's memory and allows you to manipulate the data in sophisticated ways. Let's see how this works in a program.

```
10 PRINT "HOW MUCH SOFTWARE DO YOU HAVE?"
15 INPUT N
20 DIM P(N)
25 LET TC = 0
```

```
30 FOR A = 1 TO N
35 PRINT "ENTER THE PRICE OF ONE PROGRAM";A
40 INPUT P(A)
45 LET TC = TC + P(A)
50 NEXT A
55 PRINT "YOU HAVE INVESTED ";TC;"DOLLARS IN SOFTWARE."
```

This program lets you type in the price you paid for each piece of software you have purchased. It uses the subscripted variable P(N) to store the prices. Line 10 asks you how many software programs you have. Your answer is stored as the value for the variable N in line 15. Line 20 uses the DIM keyword to tell the computer that you plan to use a subscripted variable named P in the program and that P will have a certain number of values. The number of values is determined by the value of N in this program. Since your answer to the question in line 10 is the value of N, the computer expects that number of different values for P. For example, if you tell the computer you have 10 pieces of software, the subscripted variable in the DIM statement would be P(10).

Line 25 in this program establishes the variable TC that the program will use later to keep track of the total cost of your software. Lines 30 through 55 are a FOR...NEXT loop. The first time through this loop the computer asks you to enter the price of a piece of software and assigns that price as the value of P(1). The second time through the loop it gets the value for P(2), the third time for P(3), and so on. Each time you enter a price, line 50 adds that value to TC, Total Cost of the software.

Finally, line 60 displays the sentence telling you how much money you have invested. This program illustrates one of the major advantages of subscripted variables, using FOR...NEXT loops with INPUT statements to assign values to the variables.

DIM. When you use subscripted variables in a program, the DIM instruction tells the computer what to expect. DIM instructions are generally placed at the beginning of a program so the computer knows how many values will be entered before the subscripted variable is used.

DIM is short for DIMension. When the computer comes to a DIM instruction, it sets aside a section of memory to use for storing the values of the subscripted variable. If you do not use a DIM statement before the subscripted variable appears in the program, the ST will automatically set aside space for 11 values--0 through 10--of the subscripted variable.

Omitting DIM statements will cause no problems unless you try to enter more than 11 values for the subscripted variable. If you try to enter too many values or use the same variable name in two DIM statements in your program, the computer will generate an error message.

The subscripted variable in this example program is a one-dimensional array like our one row of seats in the auditorium. You may use as many one-dimensional variables as the memory of your computer will allow. Usually you will have plenty of memory for all the variables you might need in your program. If you should happen to run out of memory, the computer will let you know by displaying an error message.

You can also use variables with more than one subscript in your programs. Let's go back to the "seats in the auditorium" example.

You know from buying reserved seat tickets to a concert or play that your seats are in one row of an auditorium full of rows of seats. Instead of one row, the auditorium is made up of sections with row after row of seats side by side. Each section of seats would be a matrix or two-dimensional array. The first seat in the

first row might be given the variable name A(1,1). (The seats in other parts of the auditorium might be in array B, C, D and so on.)

Double subscripts tell the computer the column and row placement of the variables. DIM is also used to set up these variables. DIM A(3,2) sets aside enough memory for a variable that has six values, two in each of three rows:

```
A(1,1) A(1,2)
B(2,1) B(2,2)
C(3,1) C(3,2)
```

You can perform mathematical computations with these numeric subscripted variable values just as you can with other numeric variables. A statement like A(1,1) + A(2,2) + A(3,2) is perfectly acceptable in a BASIC program.

DIM and string variables. Suppose you want to use words instead of numbers as values for your subscripted variables. BASIC will allow you to use string subscripted variables. These variable names must include $. You might use NAME$(1) or ADDRESS$(5) in a computerized address book program.

Printing Your Program

Printed output from your computer is called hardcopy. You can get a hardcopy of any of your BASIC program listings by telling your computer to output information to the printer instead of to the screen.

GRAPHICS AND ST BASIC

The ST is well endowed when it comes to graphics and color, but accessing those features through ST BASIC is a bit like having a sleek new Italian sports car that you can drive only if you hitch a mule up to the front bumper to pull it through town. Some of the power and beauty of ST graphics is lost, or at least difficult to find, when you work in BASIC. If you are interested in exploring graphics we recommend some of the graphics programs reviewed in Chapter 9. Software such as *DEGAS* from Batteries Included and *Easy Draw* from Migraph, and *NEOchrome* from Atari are all outstanding products that let you harness the power of ST graphics without problems and the limitations of ST BASIC.

BASIC, however, is provided free with every ST, and graphics programs such as Degas cost extra. With that in mind, this section introduces the graphics commands available in ST BASIC.

Graphics Modes on the ST

The ST has three graphics modes that were discussed in Chapter 2. In the high resolution mode, it can display monochrome graphics that are made up of pixels arranged in 616 rows of 345 pixels. This 616 by 345 HIRES mode allows you to individually control 212,520 picture elements (pixels)! This mode is available only if you have an Atari monochrome monitor (SM124).

With the SC1224 color monitor, you can use a medium resolution mode that displays 608 rows of 167 pixels in up to four colors or a low resolution mode of 304 rows of 167 pixels in up to sixteen colors. You can select the resolution mode in the Set Preferences submenu of Options on the GEM desktop.

Any of the three resolutions allow you to individually control the thousands of individual dots or pixels that can be displayed on the screen.

Dots or pixels are also called *bits* because each dot on the screen is controlled by one bit in the computer's memory. If a particular bit is a 1, the dot it controls will be on or displayed. If the bit is 0, that dot will not be lit on the screen. When you select one of the three resolutions, the computer sets aside a section of memory to hold the information that controls which dots are on and which are not. In the high resolution mode, the bit that controls a particular pixel simply determines whether the dot is on or off. All the dots will be the same color. If you want different dots to be different colors, that requires additional memory since the computer must keep track of a dot's color as well as whether it is on or off.

The ST sets aside enough memory to display 101,536 pixels in any of four different colors in the medium resolution mode (608 by 167). It can only keep track of half as many pixels, 50,768, if you want the option of using up to 16 different colors in the low resolution mode (304 by 167).

The Screen Bit Map

When you work with individual pixels on the screen, that is referred to as "bit mapped graphics". The screen is divided into rows and columns of dots. A particular pixel can be specified by two numbers, the X and Y coordinates for that location. The X coordinates are along the horizontal screen axis. The leftmost column on the screen is X coordinate 0. The rightmost column is X coordinate 166 in the low and medium resolution modes, and 344 in high resolution mode. Remember that the first position is assigned the number 0 which means the 167th position is position number 166.

The Y coordinates are along the vertical screen axis. The top row on the screen is Y 0 and the bottom row is Y 303 in low resolution, 608 in medium resolution, and 616 in high resolution. The top left corner of the screen is thus 0,0 (the X coordinate is listed first in commands), and the bottom right corner is 607,166 in the medium resolution mode. Many of the graphics commands in ST BASIC use the X and Y coordinates to indicate where a figure is to be drawn.

COLOR

This keyword lets you tell the computer which colors to use for items such as text displayed on the screen, lines drawn, and shapes that are to be filled with color. The general format for the command is:

```
COLOR text#, fill#, line#, style#, index#
```

The text number determines the color of any text printed on the screen. In low resolution mode, text# can be a number between 0 and 15 since any of sixteen colors can be used. In medium resolution mode, text# can be any number between 0 and 3 since four colors are available. Unless you change the control panel on the GEM desktop, the following colors are assigned to the numbers:

0 - White	4 - Blue	8 - Grey	12 - Hot Pink
1 - Black	5 - Dk Blue	9 - Lt Black	13 - Purple
2 - Red	6 - Dk Red	10 - Aqua	14 - Gold
3 - Green	7 - Lt Green	11 - Bl.Grn	15 - Yellow

The BASIC command, COLOR 2, tells the computer to display any text in red, while COLOR 14 tells it to output text in a color that looks a bit like gold after it has been drug through the mud.

You might think of COLOR 0 as an instruction to make text invisible since the default background color is white, and any text you output in that color will not be visible unless you change the background color.

The set of numbers listed is also used to indicate the fill color and the line color. For example, the command COLOR 8,4,2 tells the computer to output grey text, to fill shapes with blue, and to draw lines in red.

Keep in mind, however, that the colors associated with the numbers are determined by the Control Panel of the Desk menu on the GEM desktop. You can adjust the Control Panel to assign any number to any color.

The other two arguments for the COLOR keyword are Style and Index. These two do not deal with colors. The Style number can be any number from 0 to 2. If Style is 0, shapes are hollow, if Style is 1, shapes are filled with a pattern, and if Style is 2, shapes are filled with crosshatching. The Index number determines the type of crosshatching used. It can be any number from 1 to 24.

The use of Style and Index will be illustrated later when the CIRCLE keyword is discussed.

CIRCLE

The CIRCLE command can be quite useful for drawing geometric shapes. You can use this keyword to draw circles and arcs. The basic format for the CIRCLE command is:

```
CIRCLE Xcenter, Ycenter, radius, start angle, end angle
```

The Xcenter and Ycenter numbers tell the computer where the center of the circle will be located on the screen. The radius number indicates how large the radius of the circle will be in pixels. For example:

```
CIRCLE 150, 50, 45
```

will draw a circle about three inches across in low resolution mode that is near the top center of the screen.

If you provide numbers only for Xcenter, Ycenter, and radius, a full circle will be drawn. If you want to draw only part of a circle instead of the entire circle, you must provide numbers for start angle and end angle. Think of the right of the screen as due East or 0 degrees, the top of the screen, or due North is 90 degrees, due West or left is 180 degrees, and South or down is 270 degrees.

A full circle thus goes from 0 to 359 degrees. You can define the location and size of the "arc" you want to draw with the start and end angle numbers. Start angle and end angle are expressed in tenths of degrees, so a start angle of 90 degrees would be expressed as 900 while an end angle of 180 degrees would be expressed as 1800. Here is the instruction:

```
CIRCLE 150, 50, 900, 1800
```

The CIRCLE 150, 50 instruction produces an open circle. If you want the circle filled with color, that must be specified with the COLOR keyword. For example, COLOR 2, 4, 15 says print text in red, fill shapes with blue, and draw lines in yellow. If that COLOR instruction were issued before a circle command, it would be drawn in yellow since the CIRCLE keyword draws the outline of a shape.

FILL

The FILL keyword can be used to tell the computer to fill a shape with a particular color. The color used is specified by the second number in a COLOR statement. Use the CLEARW keyword to erase anything in the output window (CLEARW 2), and then type the following program:

```
10 CLEARW 2
15 FULLW 2
20 COLOR 2,4,15
30 CIRCLE 150, 50, 45
40 FOR A = 1 TO 4000: NEXT A
50 FILL 150,50
```

Line 10 clears the output window and line 15 gives that window the full screen for display. Line 20 sets the text color to red, the fill color to blue, and the line color to yellow. Lines 10 through 40 will draw the outline of a circle on the screen in yellow. The FOR...NEXT loop in line 40 is used to pause the program for a few seconds after it draws the circle. Then the fill command in line 50 fills the circle with the color blue.

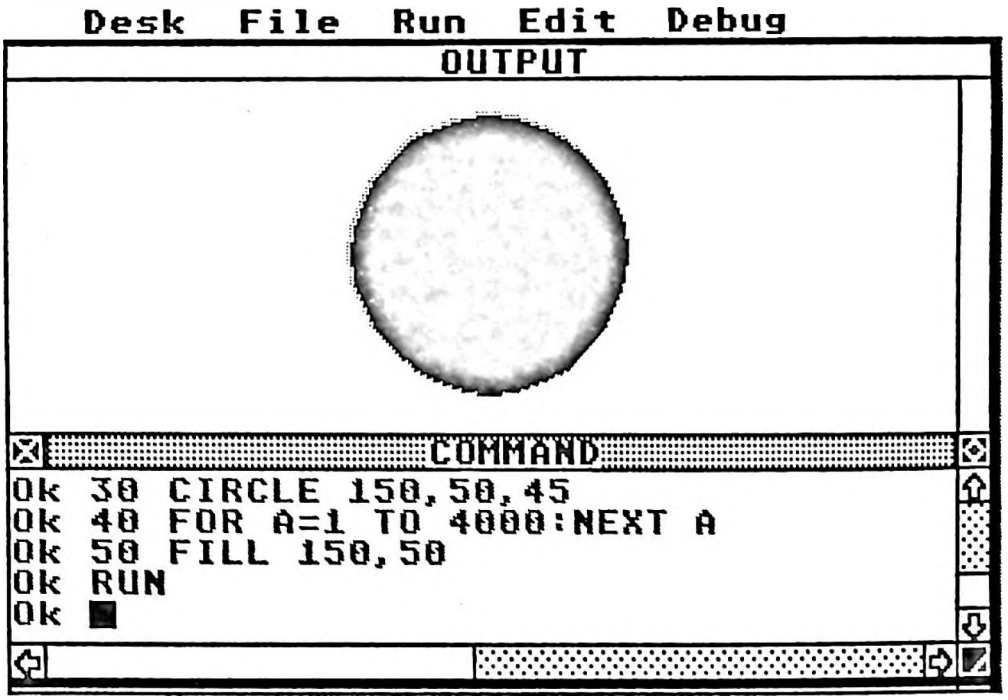


Figure 11.1-Sample program Illustrating CIRCLE and FILL Commands

FILL needs only two numbers, the X and Y coordinates, because it begins adding color to the screen at the location specified by those coordinates. It adds color until it reaches a barrier. If the beginning point is inside a shape such as a circle or rectangle, only that shape will be colored. If the beginning point is not inside a completely enclosed shape, the color will "leak out" and fill the entire screen.

PCIRCLE

This keyword is much like CIRCLE. The only difference is that PCIRCLE fills in the shape with color or shading. The general format for this keyword is:

```
PCIRCLE xcenter, ycenter, radius, start angle, end angle
```

that is, the same format as CIRCLE. CIRCLE creates the outline of a circle while PCIRCLE creates a circle filled with color. The short program below will draw a circle and fill it with blue color.

```
10 CLEARW 2
15 FULLW 2
20 COLOR 2,4,15
30 PCIRCLE 150, 50, 45
```

The second number in the COLOR statement is 4, which indicates the fill color should be blue. That is the color PCIRCLE will use to fill in the circle.

The Style and Index Arguments of COLOR

Thus far we have used only three arguments with COLOR. They determine the text color, fill color, and line color. Two other arguments, Style and Index, can be used to define the type of pattern used when shapes are filled. The program below will illustrate the way Style and Index operate.

```
20 FOR S=0 TO 3
25 FOR I=1 to 24
30 CLEARW 2
35 FULLW 2
40 COLOR 2,4,15,S,I
50 PCIRCLE 50, 75, 45
60 PRINT "STYLE IS ",S
70 PRINT "INDEX IS ",I
75 INPUT G$
80 NEXT I
90 NEXT S
```

When you run this program, the color in the circle will always be blue, but the pattern, or lack of pattern, will be determined by the value of Style (S) and Index (I). Line 75 causes the program to pause until you strike a key and press the Return key. That will give you a chance to look at the different patters produced by different Style and Index numbers. For example, when Style is 0 or 1 and Index is 4, the shading in the circle will be Atari symbols. In medium resolution mode, the circle will be open or unfilled for every combination where Index is greater than 4, but that will not be the case in the high resolution mode.

ELLIPSE

This keyword works much like CIRCLE, but it lets you create shapes that are not as uniform as those produced by CIRCLE. The general format for ELLIPSE is:

```
ELLIPSE xcenter, ycenter, xradius, yradius, starting angle, ending
angle
```

The only difference between CIRCLE and ELLIPSE is the use of two numbers for the radius, one for the x coordinate and one for y. The instruction ELLIPSE 150, 50, 20, 40 produces an ellipse that is much longer than it is wide. Remember, however, that if you set xradius to half the value of the yradius, you will not get an oval that is half as wide as it is long. That is because the bit map of the screen does not have the same number of pixels across the horizontal and vertical axes.

PELLIPSE

This keyword has the same relationship to ELLIPSE that PCIRCLE has to CIRCLE. PELLIPSE creates solid ellipses that are filled with color or shading. See the discussion of ELLIPSE for more information since the format for PELLIPSE is the same. See the section on the Style and Index arguments of COLOR for information on how to determine what color or shading is used by PELLIPSE.

LINEF

If you want to draw lines of the screen, the LINEF keyword will do that automatically. The format is:

```
LINEF xbegin, ybegin, xend, yend#
```

The first two coordinates define the beginning point, and the second two define the end of the line. For example, LINEF 0,0 303, 166 draws a line from the top left corner of the screen (0,0) to the bottom right corner (303,166) in the low resolution mode. The instruction LINEF 0,0,615,344 would do the same thing in the high resolution mode.

VDISYS(#)

The BASIC keywords discussed thus far let you draw circles, ellipses, and lines. There are also keywords in BASIC for selecting selecting the color of text and figures as well as the shading or background patterns used in graphics.

ST BASIC thus has a number of graphics-oriented commands, but they do not come close to tapping all the power of the ST. The programmers who created ST BASIC realized that and included a way for people writing programs in BASIC to tap the graphics features built into GEM's Virtual Display Interface (VDI). The VDISYS keyword lets you send instructions to GEM that tell it to create all sorts of special graphics effects such as boxes, circles, polygons, pie charts, and much more. The VDI software has many graphics subroutines that GEM uses to create the graphics displayed on the desktop. Many of those subroutines can also be put to work in a BASIC program, but the procedure for doing that is neither easy nor straightforward. When you use the keyword VDISYS(#), the computer leaves BASIC and runs a subroutine in the VDI software. (The # in the parenthesis after VDISYS is a dummy variable. You can use any number.) The particular subroutine used, and the characteristics of the graphics produced, depend on the data you supply before using VDISYS. The POKE keyword is used to "poke" the require data into the memory locations the VDI software will check before creating the graphics.

If you have the patience and desire to do it, you can use VDISYS to create many types of graphics such as dotted lines in various colors, bars for graphs,

lines of various thicknesses, pie charts, and rectangles with rounded edges. If you would like more information on VDISYS, James Luczak's article, "VDI Sampler", is an excellent place to begin. It describes the VDI subroutines and includes a BASIC program that shows how to create several types of graphics. The article was published in the June, 1986 issue of *Analog* magazine.

SOUND AND ST BASIC

The final topic of this chapter is sound synthesis, another strong point of the Atari ST. Unfortunately, it is also another strong point that is not well supported by ST BASIC. There are two keywords in ST BASIC, SOUND and WAVE, that can be used to output sound effects and music.

SOUND

The general format for SOUND is:

```
SOUND voice#, volume#, note#, octave#, duration#
```

Here is a description of the numbers you can use for each of the arguments of SOUND:

Voice - 1,2, or 3. The ST has a three voice sound synthesizer which means it can output three different sounds simultaneously. The Voice argument determines which of the voices this SOUND instruction will use.

Volume - 0 to 15. This argument determines how loud the sound will be.

Note - 1 to 12. The note argument tells the computer which note to play. Here is a chart of the notes represented by each number:

1 - C	5 - E	9 - G Sharp
2 - C Sharp	6 - F	10 - A
3 - D	7 - F Sharp	11 - A Sharp
4 - D Sharp	8 - G	12 - B

Octave - 1 to 8. The ST's sound synthesizer has an 8-octave range,and this argument is used to tell it which octave to use. Octave 1 is the lowest and octave 8 is the highest.

Duration - This argument determines how long the note will be played. Duration is given in 1/50ths of a second. Therefore,a duration of 50 is one second while a duration of 1000 is 20 seconds.

WAVE

The other sound keyword in ST BASIC is WAVE. It has five arguments:

```
WAVE enable#, envelope#, shape#, period#, delay#
```

WAVE determines the shape and pattern of the sound's envelope. Here is what each of the arguments accomplishes:

Enable - 1, 3, 8, or 15. This number determines which of the three voices is used to make the sound. 1 turns on a tone for voice 1, 3 turns on a tone for voices 1 and 2, 8 turns on the noise generator (as oppose d to the tone generator) for voice 1, and 15 turns on the tone and noise for all three voices.

Envelope - The number used here determines whether the shape of the sound produced by a voice is determined by the envelope number or by the default or standard values.

Shape - # to # - This argument controls the shape of the sound'senvelope. Some sounds start out high and drop off quickly (1), others rise quickly and drop quickly (4), Others have asawtooth pattern of rapid rise and fall across the duration ofthe sound (8 or 10). The number used for shape can be anything from 0 to 15.

Period - The number for period determines how quickly the sound's cycle or pattern is repeated. A low number means the sound will cycle quickly.

Delay - The delay number controls the time between the end of the sound and the execution of the next instruction in BASIC. Delay is given in 1/50ths of a second.

This concludes the chapters on BASIC. Although BASIC is currently the most popular programming language for beginners, there are many other languages. The next few chapters introduce several other languages that may be of interest to you.

12

Logo and Turtle Graphics

The previous chapter discussed one of the programming languages available on your ST Language Disk, BASIC. This chapter will introduce the other programming language you will find on that disk, Logo.

In many ways Logo is as much a philosophy of educational computing as it is a specific language. The language itself was developed because its creators felt existing languages did not offer children the opportunity to use computers in the most effective, most interesting ways.

This chapter explores some of the theoretical foundations for Logo and the language itself. Even though the language was designed in an educational setting it is a very powerful language with many applications.

LOGO'S ANCESTRY

Dr. Seymour Papert of Massachusetts Institute of Technology was heavily involved in creating and shaping the new language. Logo is, in fact, very much a product of Papert and his background. Son of an entomologist, Papert lived with his family in the South African bush until he was eight. Then the family moved to Johannesburg where he attended secondary school. After a brief stay at Cambridge University, he settled down to serious work at the University of Paris. After graduating with a degree in mathematics, Papert studied and worked for five years with the Swiss psychologist Jean Piaget in Geneva. The influence of Piaget's theories are obvious in Papert's writing. Those theories are part of the foundation upon which Papert builds a model of educational computing and a language--Logo.

Papert and Piagetian Theory

Piaget believed "learning-by-doing" is the basis of a child's development. Seymour Papert accepts that assumption. One of the reasons for the creation of Logo was to provide children with an interesting, electronic environment in which they could explore the concepts of mathematics, particularly geometry, by actually manipulating objects in the environment.

Piaget also believed nature sets a timetable or sequence for development of a child's thinking skills which should not, even could not, be influenced by eager parents and teachers who want to speed up the cognitive growth of children. Papert, and other supporters of Logo, reject this idea, believing instead that child-

hood experiences help determine the type and quality of the child's thinking. Papert believes appropriate experiences with Logo can lead to more effective thinking skills and problem-solving strategies.

Direct intervention, and structured attempts to teach topics selected by the teacher, are discouraged by Piagetian theory since children learn best when they discover things on their own. The leaders in the Logo movement generally support this discovery learning approach. They believe Logo classes allow children to pursue their own course, to discover things on their own with only a small amount of teacher guidance.

Papert believes the traditional approach of computer-assisted instruction is just a way of having a dumb machine, the computer, program a smart child. According to Papert, the future of educational computing lies in allowing the child to teach or program the computer.

The Logo approach creates "microworlds" where students can discover the rules governing the way that world operates. They discover those rules through manipulation of that environment. Logo also allows the child to experience directly the consequences of his or her actions. The computer is given instructions, the child observes the consequences of those instructions, and may then modify them to better accomplish a goal.

Logo is not just a language, it is an approach to educational computing that assumes the best use for computers in education is as devices children teach rather than as devices that teach children.

THE LOGO LANGUAGE

The original versions of Logo were written for large computers. They included a set of instructions children could use to guide the movement of a small mechanical "turtle," a motorized device with a pen. The turtle could be told to move in different directions, and to put its pen on the paper it was moving over to draw different figures.

Today most applications of Logo use a personal computer version of the language. Today there are at least six versions of Logo for the Apple II computer alone. Versions are also available for popular computers from Commodore, IBM, Radio Shack, and many other manufacturers. Logo was actually the first language of the Atari ST. Early versions of this computer did not come with a version of BASIC but did run Logo.

The most popular versions of Logo for small computers use a video turtle. Instead of a mechanical turtle that moves around on the floor, there is a video turtle that allows students to draw figures on a color monitor.

Atari ST Logo

Versions of Logo differ in much the same way versions of BASIC differ. There are many similarities but the version of Logo for Commodore computers is not exactly the same as the one for IBM or ST computers. The ST version comes from a company called Digital Research and is one of the more popular versions of Logo available. The same version of Logo runs on IBM computers. In addition to this chapter, any books or magazine articles which discuss the Dr. Logo version can provide information about ST Logo.

Loading ST Logo. Procedures for loading Logo from your Language Disk are the same as those discussed in previous chapters. To begin working with Logo, the LOGO.PRG. file should be loaded from your ST Language Disk.

ST Logo Windows. Once you load Logo, your screen will display the menu bar and two windows as shown in Figure 12.1. The Logo Dialogue window covers the left half of the screen. This window is much like the BASIC Command window and all the commands you type will appear in this window. The Graphics Display window covers the right half of the screen. It functions like the Output window on the BASIC screen and the picture which results from your Logo program appears in this window when the program is executed. Moving and changing the size of these windows is done with the same procedures described in the TOS chapters.

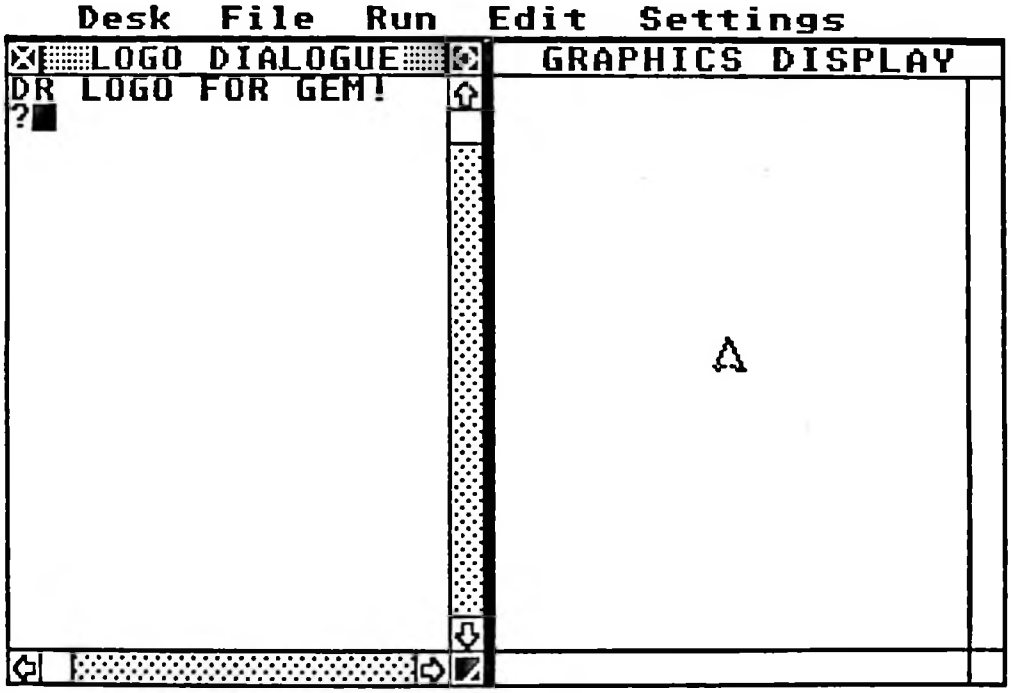


Figure 12.1-The Logo Screen.

Two other windows are available in ST Logo, the Edit window and the Debug window. These windows are discussed in later sections of this chapter.

Logo's Turtle Graphics

Although it is true that turtle graphics is only one aspect of a fully-developed version of Logo, most discussions of the language, including this one, begin with a description of turtle graphics--the approach Logo uses to introduce children to creating their own graphics. Turtle graphics is the best known aspect of Logo and the part of Logo most often taught in Logo classes. Many of the texts used in Logo classes, in fact, deal almost exclusively with turtle graphics.

In ST Logo turtle is a small triangle that appears near the center of the Graphics Display window. When you first see the turtle, the point of the triangle is pointing toward the top of the screen. This is called the "home position." The language has a comprehensive set of instructions that control what the turtle does. Keywords in Logo are called "primitives." A primitive is a word Logo already understands. Logo primitives are the equivalent of BASIC's statements and commands.

Here is a summary of four instructions used to move the turtle around on the screen:
FORWARD and BACKWARD. The point of the triangle indicates the current direction the turtle is headed. With the turtle in the home position, in the middle of the screen pointed toward the top, type

```
FORWARD 20
```

If you forget to give the computer all the instructions it needs, Logo will prompt you for the missing information. For example, if you type FORWARD and omit the number of steps the turtle is to move, a message such as TELL ME MORE may appear on the screen. And, if you leave out the space between FORWARD and 20, Logo will display an error message like the one in Figure 12.2.

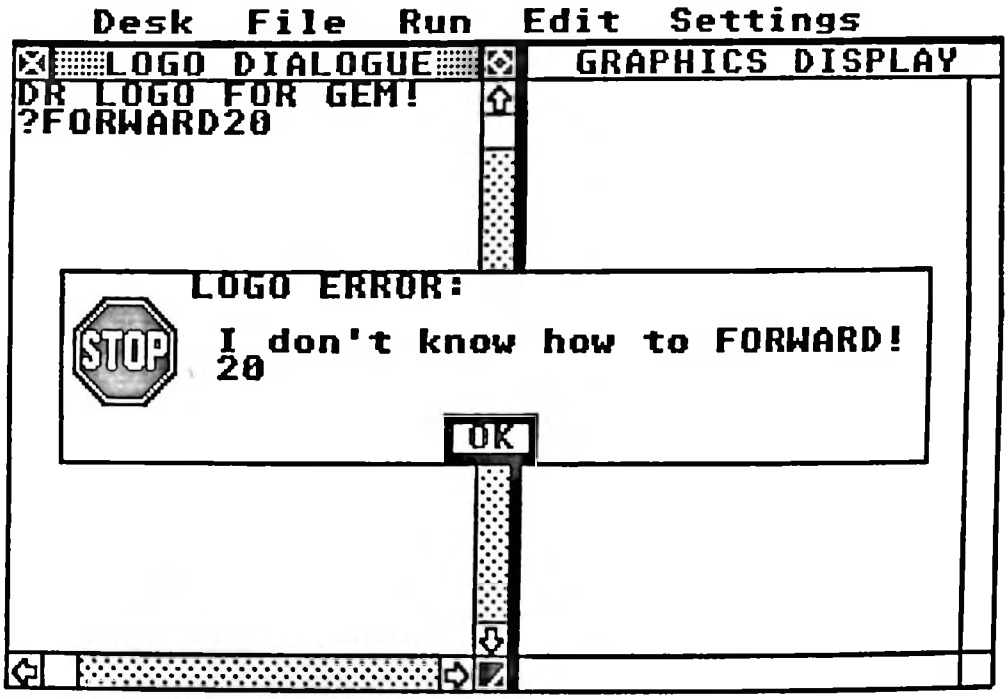


Figure 12.2-Logo error message dialog box.

When you press RETURN after typing the primitive, the turtle will move toward the top of the screen, drawing a line on the screen as it moves. FORWARD 20 tells the turtle to move FORWARD for 20 steps. A "step" is the unit of measurement in turtle graphics. The actual size of a step is only a few hundredths of an inch.

BACKWARD works the same as FORWARD except movement is in the direction opposite the way the turtle is headed. Type

```
BACKWARD 50
```

and the turtle will move BACKWARD for 50 steps. (Notice, the turtle retraces the 20 steps FORWARD from the previous instruction plus 30 more steps.)

When you use the FORWARD and BACKWARD primitives you must always tell the turtle which direction to move and how many steps to take. You can abbreviate FORWARD by typing only FD and BACKWARD by typing only BK but you must always be sure to put a space between the direction and the number of steps.
RIGHT and LEFT. These two primitives direct the turtle to rotate a certain number of degrees. Type

```
RIGHT 90
```

and you will notice the turtle turns to face the right side of the screen. Now, type

```
LEFT 90
```

and the turtle will rotate back to its previous heading.

When you use the RIGHT and LEFT primitives, you must always remember to include the number of degrees you want the turtle to rotate. Think of the turtle as being able to turn around in a complete circle. Since a circle has 360 degrees, you can instruct the turtle to rotate any number of degrees from 1 to 360. When the turtle is pointing straight up, the instruction RIGHT 90 tells it to turn and point to the right, while LEFT 90 tells it to point to the left. However, the turtle does not need to be pointing straight up when you use the RIGHT and LEFT primitives. Whenever you use these primitives the turtle will rotate RIGHT or LEFT from the direction it is headed.

With just these four primitives you can create a variety of simple shapes. The following instructions tell Logo to draw a small square on the screen:

```
FORWARD 30
RIGHT 90
FORWARD 30
RIGHT 90
FORWARD 30
RIGHT 90
FORWARD 30
```

More Turtle Graphics Primitives

Here are a few more primitives Logo understands. The abbreviations in parentheses after the primitive may be used instead of the keyword itself.
HOME. Returns the turtle to the home position from anywhere on the screen but does not erase the display. When you use this primitive, the turtle will draw a line as it moves from its location to the home position.
HIDETURTLE (HT). This primitive makes the turtle invisible. It will still carry out any instructions you enter, but it will not be visible on the screen.
SHOWTURTLE (ST). As you might expect, when you have used the HIDETURTLE to make the turtle disappear, this primitive makes the turtle visible again.
PENUP (PU). This primitive was added to Logo when the turtle was actually a mechanical device with a pen attached. PENUP told the mechanical turtle to lift its pen from the paper. An instruction such as FORWARD 20 would then cause the turtle to move, but it would not draw a line. In versions with a video turtle, the primitive works the same way. After PENUP the turtle moves when instructed to do so but does not draw a line as it moves.
PENDOWN (PD). After a PENUP primitive this one tells the turtle to start drawing lines again.

PENERASE (PE). This primitive makes the turtle erase lines you have drawn. But you must retrace the lines exactly. For instance, if you create a line with FORWARD 20, you can erase it by typing:

```
PENERASE
BACKWARD 20
```

CLEARSCREEN (CS). When your Display window gets cluttered, you can use this primitive to erase that window and send the turtle to the home position. CS does not affect the Dialogue window.
CLEARTEXT (CT). Serves a similar purpose to CS, except this one clears the Dialogue window without affecting the Display window.
FULLSCREEN (FS). This primitive is used to make the Display window full-screen size.
Setting Turtle and Pen States. The HIDETURTLE and SHOWTURTLE primitives describe the "turtle state" while PENUP, PENDOWN, and PENERASE describe the "pen state." These settings can be changed by entering new primitives or by using the Settings menu if you want to use the mouse instead of typing the primitives. Move the mouse pointer to the Settings menu, choose the TURTLE option on that menu and you will see the Dialog Box containing the choices for Turtle Settings. This Dialog Box is shown in Figure 12.3.In addition to Turtle State and Pen State settings, you can set the Turtle Position. We will discuss Turtle Position later in this chapter.

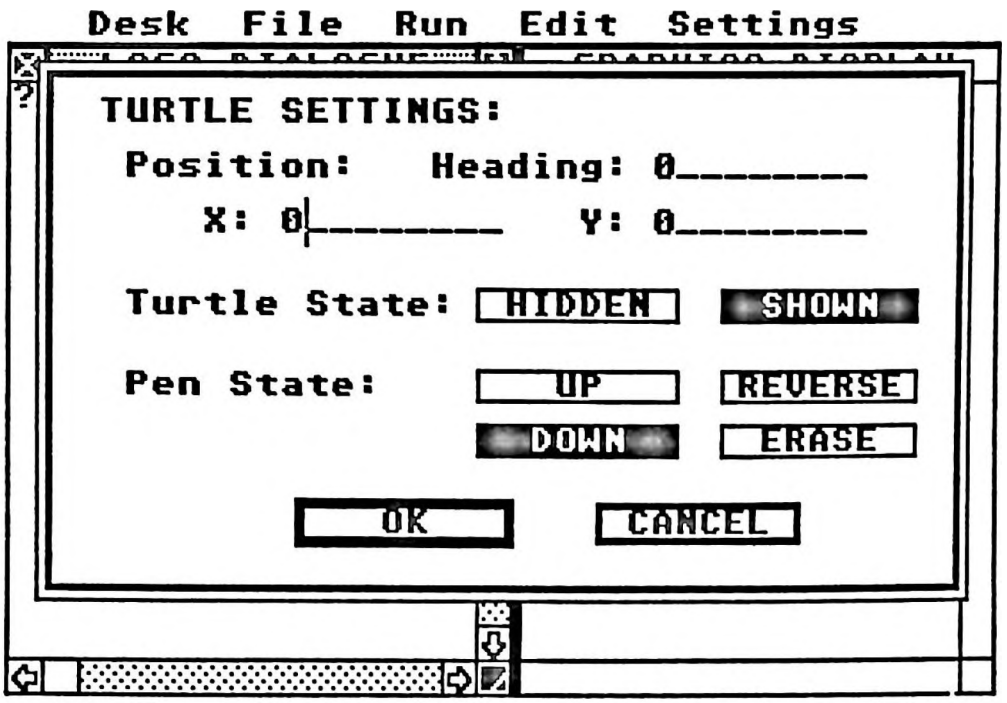


Figure 12.3-Turtle and pen state settings dialog box.

REPEAT. Creating graphics often involves repeating the same commands. For example, the program that drew a box used FORWARD and RIGHT four times. The REPEAT keyword can often eliminate repetition in your program. The instruction

```
REPEAT 4[FORWARD 30 RIGHT 90]
```

also draws a box. The instruction tells the computer to repeat the instructions inside the brackets four times. You could have more instructions inside the brackets. Or, you could repeat the instructions any number of times.

THE SETTINGS MENU

ST Logo's Settings Menu, shown in Figure 12.4, allows you to adjust three different kinds of settings: Graphics, Turtle, and Screen. When you select either of these options on the menu a Dialog Box is displayed. You use the mouse or the keyboard to change any of the settings, then click on the OK box to let Logo know you are finished. (Pressing the RETURN key has the same effect as clicking on the OK box.) We will use two more primitives, BOX and CIRCLE, to illustrate how all these settings work. Since you have already been introduced to Logo's turtle we will look at the Turtle Settings option on this menu first.

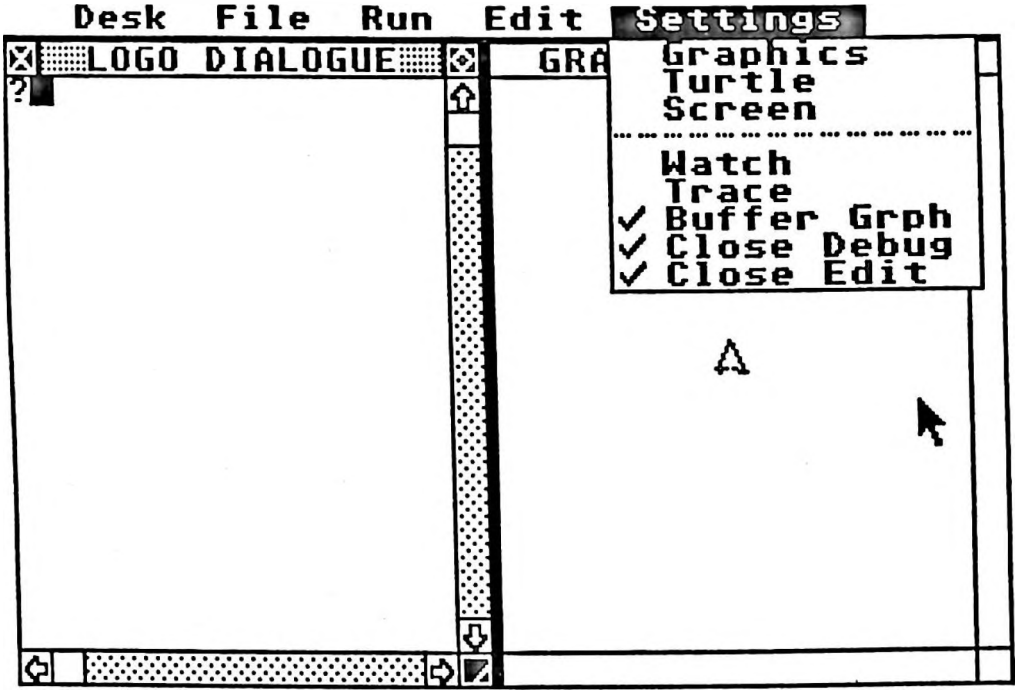


Figure 12.4-The Logo settings menu.

Turtle Settings

You saw the Turtle Settings Dialog Box earlier in Figure 12.3 when we discussed turtle and pen state. Now we will use this same Dialog Box to set the turtle position.

Turtle Heading. One important part of the turtle's position is the direction in which it is pointed, the "heading." If the turtle Heading is 0, the turtle is pointed toward the top of the screen. Heading 90 is toward the right side of the screen. Heading -

90 is toward the left of the screen. You set the turtle's heading by entering a positive or negative number between 0 and 360.

Enter 90 as the Turtle Heading then click on the OK box (or press the RETURN key). Notice the turtle is now facing the right side of the screen. Change the heading to -45 and you will see the turtle turn a different direction.

The number you are entering as the heading is actually the number of degrees you want the turtle to rotate. Positive values cause the turtle to turn clockwise from the 0 heading; negative values, counter-clockwise from the 0 heading.

You can also control the turtle's heading with the SETHEADING (SETH) primitive. The format for this primitive is

```
SETH # of degrees
```

where # of degrees is a positive or negative number between 0 and 360.

You may be asking how setting the heading in the Dialog Box or using the SETH primitive is different from using the RIGHT and LEFT primitives. RIGHT and LEFT primitives cause the turtle to rotate a specified number of degrees in relation to the direction the turtle is pointed regardless of its heading. Using either method to set the heading differs from using RIGHT and LEFT in that the number of degrees specified is in relation to the 0 heading, not to the direction the turtle is pointed. The following uses SETH to illustrate the difference. Press the RETURN key after each line and watch what the turtle does.

```
HOME
RT 45
SETH 90
FD 50
HOME
RT 45
RT 90
FD 50
```

We used the HOME primitive to make sure the turtle was in the home position (e.g., heading 0). Also we used the abbreviations for the FORWARD, RIGHT, and SETHEADING primitives. The second line of this program causes the turtle to rotate 45 degrees to the right. The turtle heading is now 45. SETH changes the turtle's heading to 90--pointing toward the right side of the screen--because SETH 90 rotates the turtle 90 degrees clockwise *from the 0 heading*. FD 50 caused the turtle to draw a line 50 turtle steps long.

Then, we returned the turtle to its home position with the HOME command. Again, RT 45 rotated the turtle 45 degrees to the right--turtle heading 45. But, this time we used the RT 90 instruction instead of SETH 90. When the turtle rotates 90 degrees to the right in carrying out this instruction the turtle heading becomes 135--not 90 as when the SETH primitive was used--because RT 90 rotates the turtle 90 degrees *from the previous turtle heading* (45). Again FD 50 caused the turtle to draw a line 50 turtle steps long. You can see the difference between the results in Figure 12.5. Since the two lines the turtle drew are in different positions on the screen.

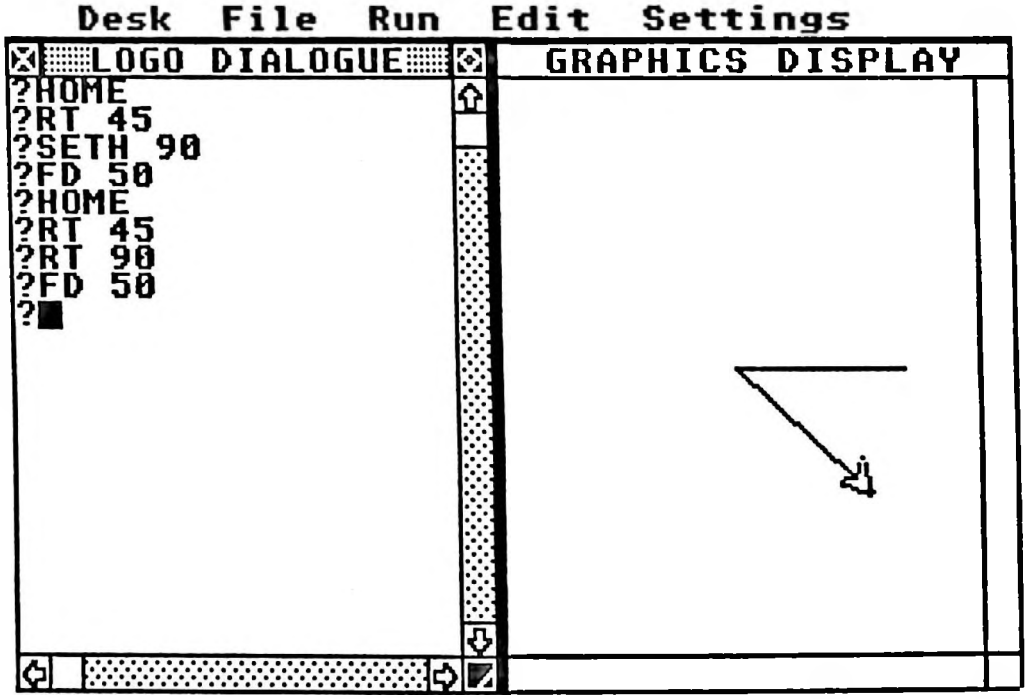


Figure 12.5-The lines resulting from Logo commands.

Turtle Position and Screen Coordinates. The Logo Display Graphics window can also be compared to a graph with an x axis and y axis. These lines cannot be seen on the screen but they cross in the center of the Display Graphics window with the x axis running from left to right and the y axis running up and down.

Points on the screen can be identified as x,y coordinates. The turtle's home position, the center of the screen, is coordinate 0,0. X coordinates to the right of the center are positive values; to the left of center, negative. Y coordinates from the center toward the top of the screen are positive values; from the center toward the bottom, negative. You can set the turtle's position by entering values for the x and y coordinates in the Dialog Box. Enter 35 for the x coordinate and 50 for the y coordinate, then watch what the turtle does. It moves toward the upper left corner of the screen, drawing a line as it moves. The point where the turtle stopped is screen coordinate 35,50.

You can also set the turtle's position from the keyboard by using the SETX and SETY primitives. Type the following primitive:

```
SETX 25
SETY -55
```

Figure 12.6 shows the result when you press RETURN after typing these primitives. The turtle moves to the lower right hand part of the screen in the position of the x,y coordinate 25,-55.

Default values for the x and y coordinates of the turtle position is 0, the home position. Typing CS to clear the screen causes the values of the x and y coordinates to return to 0 even if you have changed them in the Dialog Box.

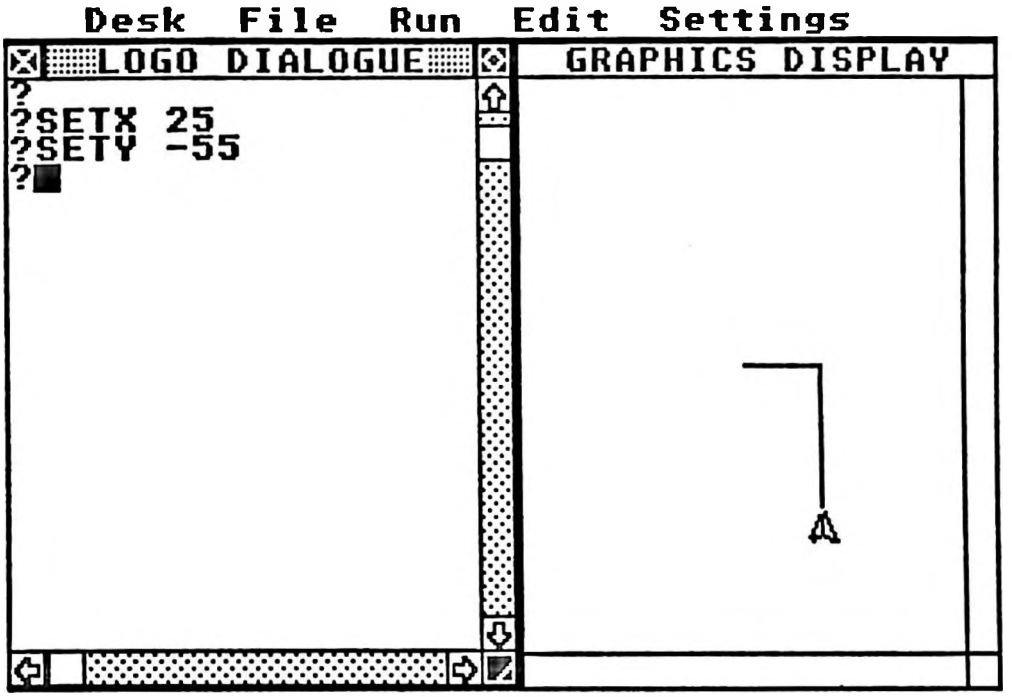


Figure 12.6-Lines resulting from SETX and SETY.

Graphics Settings

Patterns used to fill designs drawn with Logo primitives, width and style of the line the turtle draws, and screen colors can all be changed when you choose this menu option. Let's experiment and run a primitive several times to see what effect different settings have on the result.

Fill options. First, choose the Graphics option on the Settings menu. The Graphics Settings Dialog Box will appear. Now, click the left mouse button on the TRUE box to tell Logo you want to FILL the figure your primitive draws. Style, Index, and Color settings determine how the figure will be filled.

Style is one of the parameters which determines the fill pattern. The possible values and their corresponding pattern are: 0, Hollow; 1, Solid; 2, Pattern; 3, Hatch; 4, User-Defined. The default value is 0 so, unless you change it, the figures drawn with primitives will be outlined. Style 1 will cause Logo to fill the figures with a solid color. Styles 2 and 3 can be used in combination with different Index values to produce a variety of line patterns inside the figures drawn. And Style 4 allows you to make up your own pattern. Index applies only when you are using Style 2 or 3.

We will use two new primitives, BOX and CIRCLE, to try out the different patterns. Move the cursor to the Style setting and enter 2 from the keyboard. Then move to the Index setting and enter 3 from the keyboard. These are the only two settings we are going to change, so click on the OK box or press the RETURN key. Then type this primitive:

```
BOX [10 0 60 75]
```

The BOX primitive used here has four arguments: x and y coordinates, width of the box, and height of the box. In this example, the box will be drawn be-

ginning at x coordinate 10, y coordinate 0. The box drawn will be 60 turtle steps wide and 75 turtle steps high. The values you entered for the Style and Index settings cause the box to be filled with the pattern you see. You can try changing these values to see what effect they have on the fill pattern. If you have a monochrome monitor, this setting will be either 0 (black) or 1 (white). If you have a color monitor you can also change the Color setting to see the box in different colors. The available colors depend on the settings on the Control Panel and the screen resolution you are using. Chapter 4 gives you details on setting these colors. All you need to do on the Color setting is type in the number of the color you want the box to be.

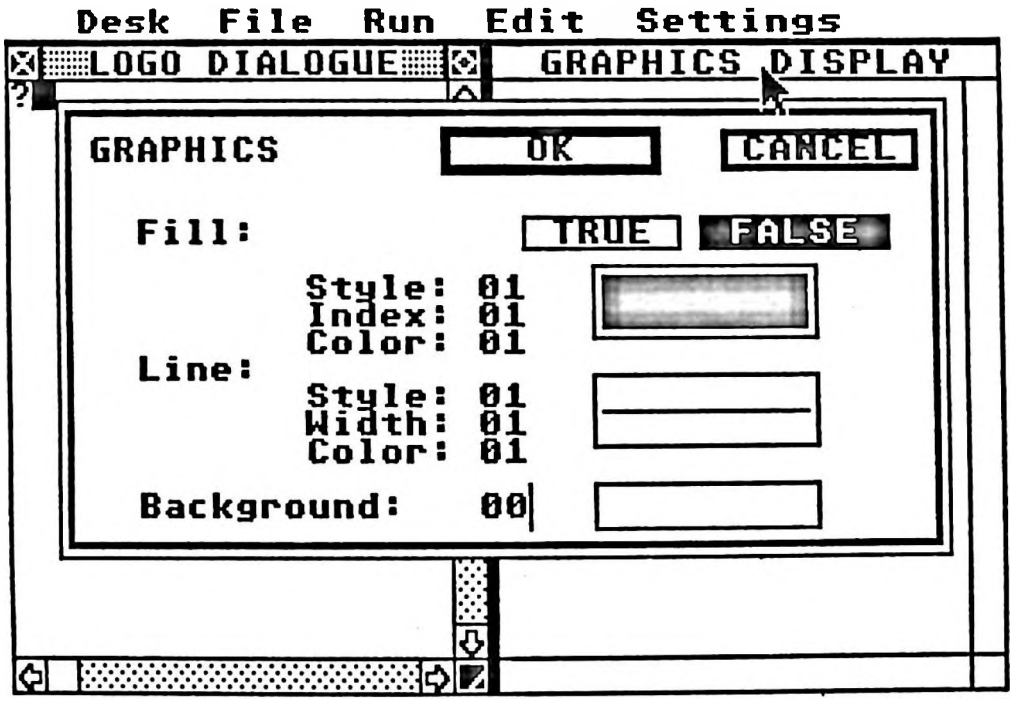


Figure 12.7-The graphics settings dialog box.

All of the Fill parameters can also be set with the SETFILL primitive. The SETFILL primitive takes this format:

```
SETFILL [style# index# color#]
```

where style#, index#, and color# have the same possible values you might enter in the Dialog Box.

Line Settings. These settings are used to establish the type and width of the line drawn by the turtle when it moves around the screen. Style is a value from 1 to 7 with 7 being the user-defined style. Width allows for any value from 1 to 39 but it doesn't work when the Fill choice is set to True. In other words, the line width will always be the same when you are filling the figures.

Before you try the next example, click on the box which says False in the Graphics Settings Dialog Box to turn off the pattern. Type this primitive:

```
CIRCLE [-20 30 40]
```

The CIRCLE primitive takes three arguments: x and y coordinates and length of the radius of the circle. This example causes the turtle to draw a circle beginning at screen coordinate -20, 30. The radius--the distance from the edge to the center of the circle drawn--will be 40 turtle steps.

SETLINE can also be used to establish all the settings for the line. This primitive must include three parameters in this format:

```
SETLINE [style# width# color#]
```

Background Color. This is the last graphics setting available. You can change the background color by entering different number values in the same way you set Fill and Line colors. Or, you can enter the SETBG primitive from the keyboard. The SETBG primitive must be in this format:

```
SETBG color#
```

where the value for color# again depends on the screen resolution you are using and whether or not you are using a monochrome or color monitor.

SCREEN SETTINGS AND OTHER MENU OPTIONS

These settings will be discussed in the next section of this chapter as they are applied to graphics examples.

Primitives and Procedures

Logo programming is not limited to primitives. One of the language's most powerful features is the ability to be taught new words by the programmer. Logo can be taught new keywords which are called procedures. Suppose, for example, that the program you are writing requires you to draw several triangles. Using the REPEAT command, you can draw a triangle with one line:

```
REPEAT 3[FD 50 RT 120]
```

Since TRIANGLE is not a primitive Logo understands, you would need to type in this line at each point in your program where you wanted to draw a triangle. However, there is a much easier way to accomplish drawing several triangles in your program since you can teach Logo to make triangles by writing a TRIANGLE procedure. To do that you type:

```
TO TRIANGLE
REPEAT 3 [FD 50 RT 120]
END
```

The primitive TO tells Logo you want to create a procedure. The name of the procedure will be the word following TO, in this case TRIANGLE. You then type in the lines that make up the procedure. END tells Logo you have finished defining the procedure. As long as that procedure is in the memory of your computer, Logo will carry out all the instructions in the procedure every time you type the name of the procedure. In other words, since you have defined TRIANGLE, Logo will perform the REPEAT instruction every time you type TRIANGLE. If you typed TRIANGLE without defining the procedure, Logo would respond with I DON'T KNOW HOW TO TRIANGLE.

The TRIANGLE procedure was relatively simple but procedures can be very complex. You can use any name you wish for a procedure as long as it is not a Logo primitive. Procedures may include any number of lines. They can be run as soon as they are defined to see if they produce the desired result. If your procedure doesn't work, it can easily be edited or modified until it is correct.

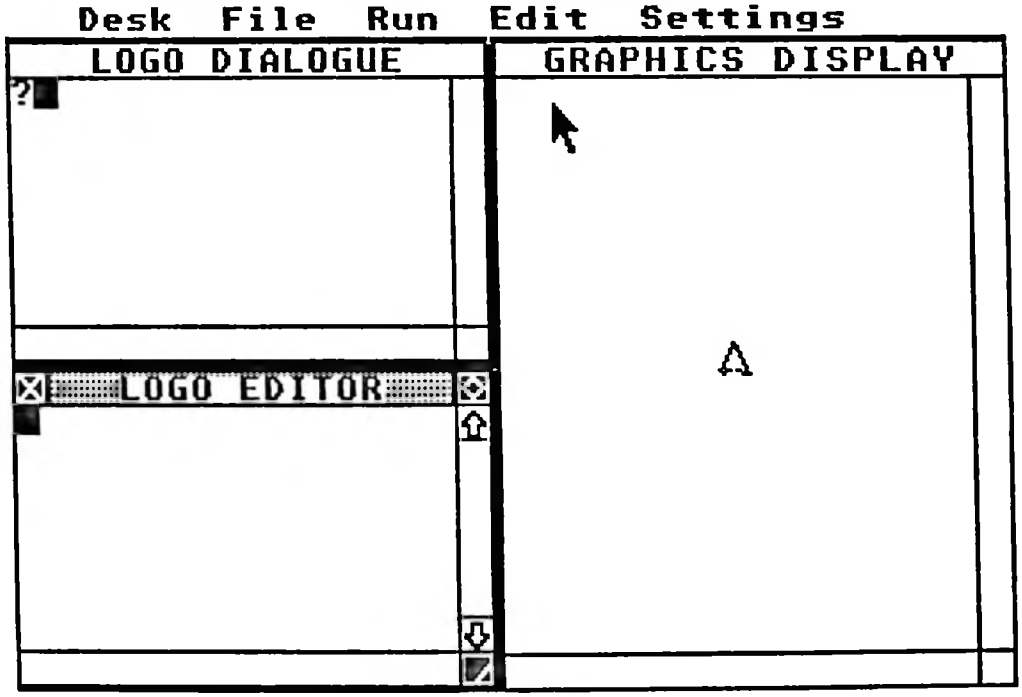


Figure 12.8-The Logo screen with the editor visible.

Editing Logo Procedures

If your procedure contains some errors or doesn't do what you anticipated it would, you can use the Logo editor to make the needed changes. The Edit menu allows you to make corrections or additions to a procedure. Choosing Workspace on the Edit menu will display a listing of all the procedures in the computer's memory. If you want to edit a particular procedure, you can type ED, quotation marks, and the name of the procedure to display only that procedure (e.g., ED "TRIANGLE). The Editor allows you to mark sections of your procedure, insert another copy of the marked section in another place in the procedure, or "cut" a section from one part of your procedure and "paste" it into another place in the procedure.

The Editor also allows you to move the cursor to the beginning or end of the procedure and move it up or down one page or line at a time. The Edit menu displays all the editing options. You can choose the options with the mouse or enter them from the keyboard. To enter them from the keyboard, you simply hold down the Control key and press the letter key shown on the menu. For instance, typing Control/W will delete a section of your procedure after it has been marked. And Control/X moves the cursor to the last line of your procedure.

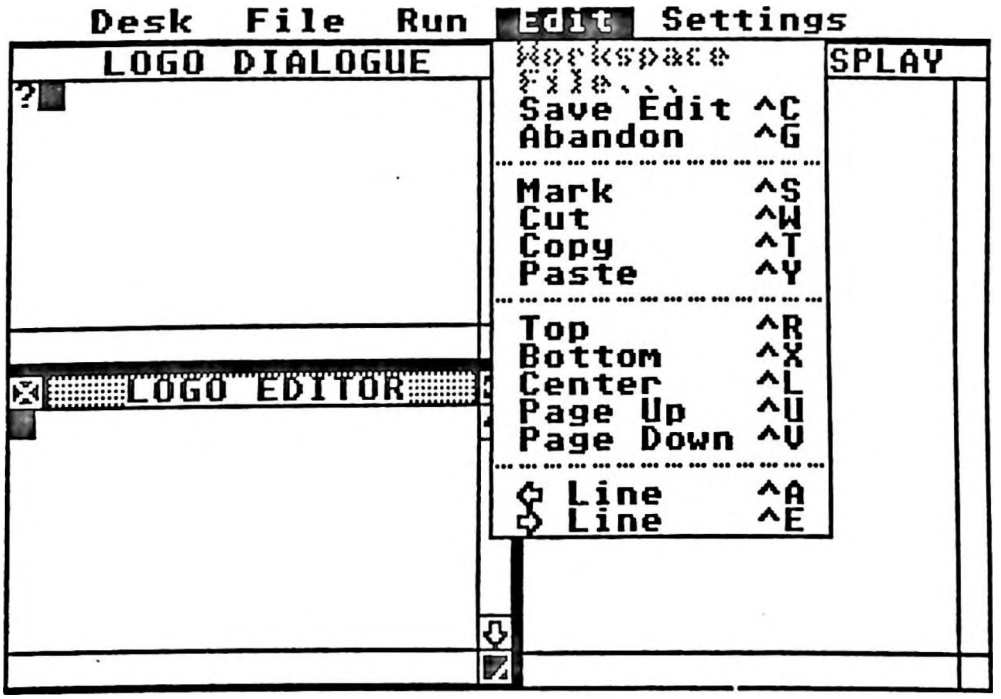


Figure 12.9-Logo Edit Menu.

Using Procedures in Other Procedures

Logo procedures allow you to write programs in sections. Any procedure you define can be used in other procedures. The following examples will give you an idea of the way it works. First define this BOX procedure:

```
TO ONESQUARE
REPEAT 4 [FD 25 RT 90]
END
```

The ONESQUARE procedure draws a square. Once you have defined the procedure, all you need to do to get Logo to draw the square is type ONESQUARE. The turtle will draw a line 25 steps long, turn 90 degrees to the right, draw another line 30 steps long, turn 90 degrees again, and so on until a square is drawn.

More Procedures. Since you defined the ONESQUARE procedure, it is still in the memory of your computer. Now, define this additional procedure:

```
TO SQUARE :SIZE
REPEAT 4 [FD :SIZE RT 90]
END
```

This procedure has an added ingredient. It requires an input. You can't simply type SQUARE and execute the procedure. Notice the :SIZE in the name of this procedure. SIZE is a variable in this procedure. To include a variable in a procedure, you must put the variable name in the procedure. The : tells Logo that what follows is a variable name. When you execute the procedure you must give the variable a value. For example, if you typed SQUARE 40 after defining this proce-

dure, Logo would execute the procedure with 40 as the value of the variable SIZE and draw a square with each side 40 turtle steps long.

Now let's define a procedure that "calls", or uses, both the ONESQUARE and SQUARE :SIZE procedures.

```
TO BOXES
REPEAT 4 [ONESQUARE RT 90]
HOME
REPEAT 4 [SQUARE 40 RT 90]
HOME
REPEAT 4 [SQUARE 60 RT 90]
END
```

Now type BOXES to run this procedure. The first line of the procedure causes Logo to REPEAT executing the ONESQUARE procedure and turning the turtle 90 degrees to the right 4 times. The second REPEAT instruction tells Logo to execute the SQUARE procedure with 40 as the value of the variable SIZE and turn right 90 degrees. Again, Logo executes these primitives 4 times. The final REPEAT instruction again calls the SQUARE procedure but this time the value of the variable SIZE is 60. We included HOME between the REPEAT instructions so that the turtle would begin from the same position and heading before executing the REPEAT instructions.

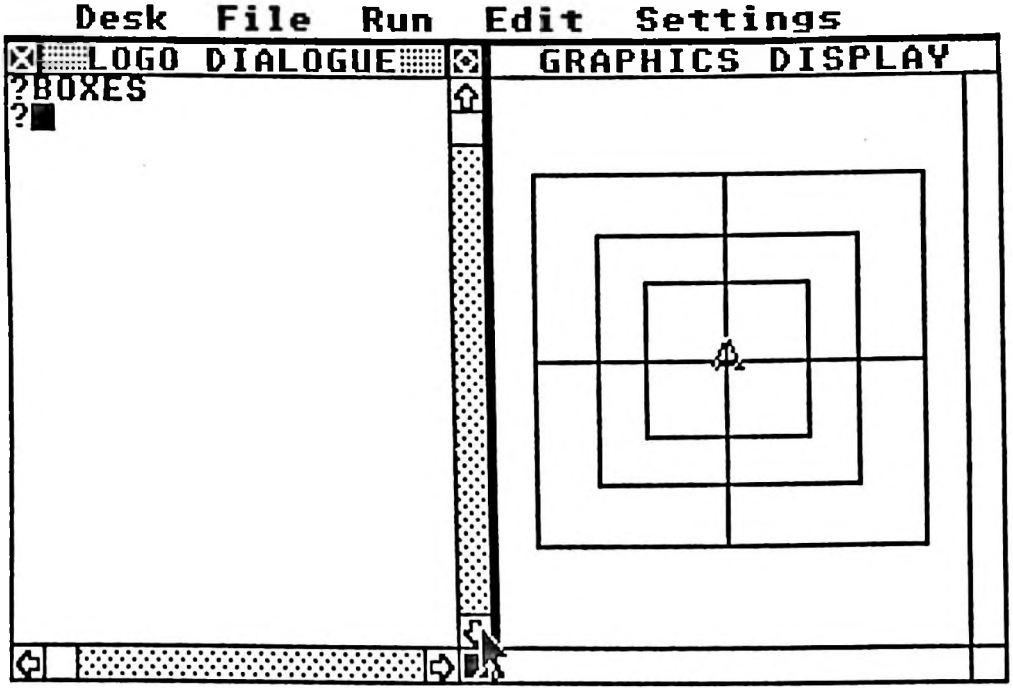


Figure 12.10-The result of the BOXES procedure.

If you would like to see the steps of your procedure as they are executed, choose the Watch or Trace option on the Settings menu. Watch opens the Debug window in the lower left corner of the Logo desktop and displays each step of your

procedure in that window as it is executed in the Graphics Display window. Trace also opens the Debug window and displays each step of your procedure as it is executed.

Trace is different from Watch, however, because Trace shows the values assigned to variables and the primitives involved in each step of the procedures. Watch merely tells you which procedure is being executed with the message Evaluating procedure name. WATCH and TRACE are also primitives which can be entered from the keyboard. To turn off TRACE or WATCH you simply type NOTRACE or NOWATCH.

Recursion: Another Powerful Logo Feature

Another programming process, recursion, is also available in Logo. The previous example illustrated one procedure calling another procedure. Recursion allows you to write a procedure that calls itself. Here is an example:

```
TO SPIRALS :SIZE
FD :SIZE
RT 75
SPIRALS :SIZE + 2
END
```

This is a simple procedure but it has a fascinating result. Type SPIRALS 1, press RETURN, and watch what happens. When you type SPIRALS 1, 1 becomes the value for :SIZE, the variable which tells Logo how many FORWARD steps to take. In the first line of the procedure, the FORWARD primitive causes the turtle to take 1 step forward. Then the turtle turns right 75 degrees. Then, the SPIRALS procedure calls itself. But this time the variable SIZE is equal to SIZE + 2. In other words, the value for SIZE is now 3. The SPIRALS procedure is executed

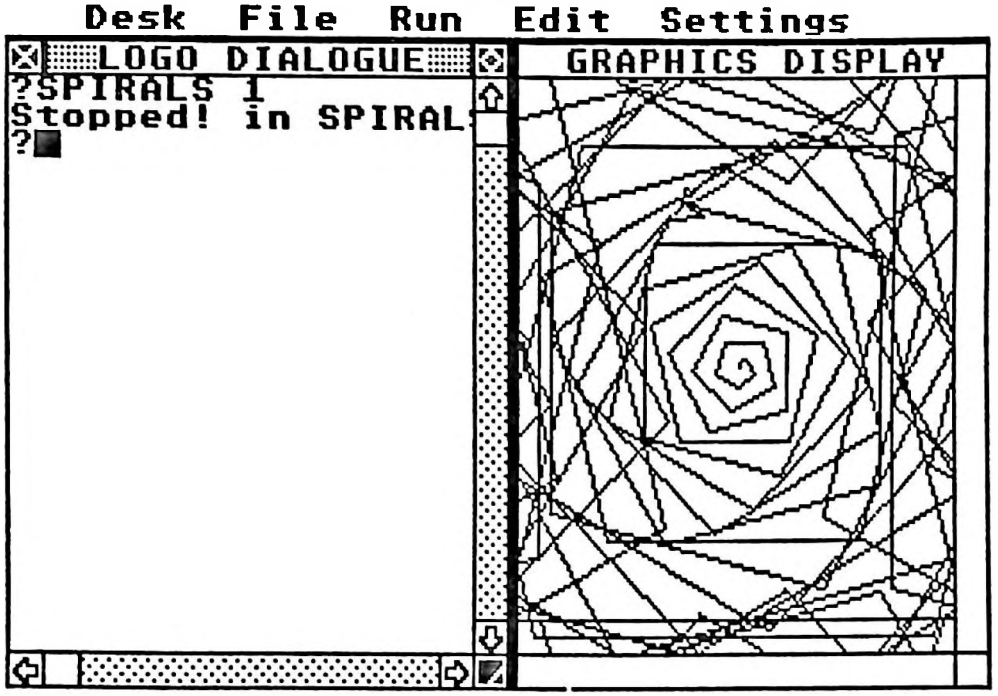


Figure 12.11-Result of the SPIRALS procedure.

with this value. The procedure continues to execute with the value of SIZE increasing by 2 each time until you stop the procedure.

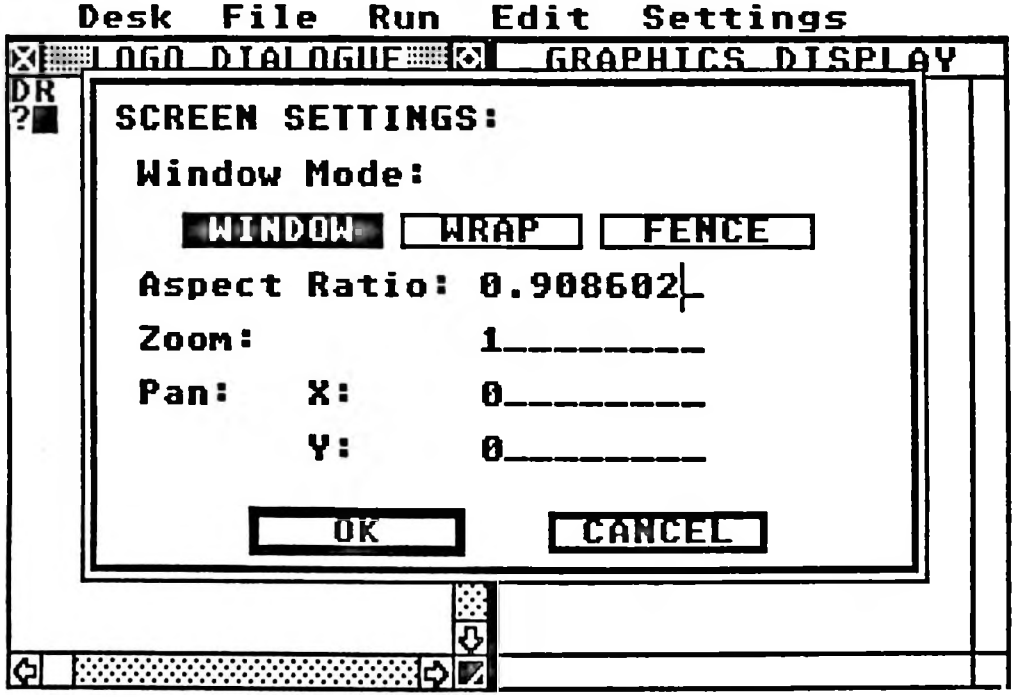


Figure 12.12-Logo screen settings menu.

Stopping Procedures and Setting Turtle Boundaries. You can stop this or any other procedure by selecting the Stop! option on the Run menu at the top of the screen, or by typing Control/G--holding down the Control key and pressing G.

You can also use the WINDOW, WRAP, and FENCE primitives to set boundaries for the turtle. WINDOW lets the turtle draw beyond the edge of the window. When you choose the WRAP option, the turtle draws off one edge of the window and reappears on the other side of the window. The FENCE primitive sets the turtle's boundaries at the edges of the window. If your procedure causes the turtle to go off the edge of the window, Logo displays a TURTLE OUT OF BOUNDS message. The WINDOW, WRAP, and FENCE primitives may be typed in from the keyboard. If you choose the Screen option on the Settings menu, a Dialog Box appears allowing you to choose one of these primitives with the mouse.

Other Screen Settings. The Screen option on the Settings menu also allows you to set some other parameters which affect the way your graphic picture looks. The Aspect Ratio, Zoom, and Pan settings allow you to adjust the size and screen placement of the pictures your procedures display. You may want to experiment with these settings to see how they affect your graphics display.

Logo and Words

Thus far the discussion of Logo has concentrated on Logo's turtle graphics. The language also has many primitives that let you deal with both numeric values and text. The ST version can perform basic math operations such as 36/6 (36 divided by 6) with whole numbers and decimal numbers up to seven digits. This version also has primitives for dealing with trigonometric functions such as sine, cosine, and tangent.

ST Logo also has primitives that let you deal with words and relationships between words. The Logo language is actually a derivative of LISP, a language frequently used in artificial intelligence research. LISP stands for LISt Processing Language. Like LISP, Logo has many ways of handling lists of both numbers and words.

Displaying Text. You can use three different primitives to display text on the screen. The text to be displayed is included in procedure lines as a list in the following manner:

```
TO WORDS
SHOW [WHAT WHEN WHERE WHY]
TYPE [THIS IS A DEMO.]
PRINT [1 2 3 4]
PRINT [NEXT LINE]
```

The material inside brackets in each of these lines is a list. When you execute these lines, you should see

```
[WHAT WHEN WHERE WHY]
THIS IS A DEMO. 1 2 3 4
NEXT LINE
```

in the Logo Dialogue window. Since these are not graphics primitives, the result of executing this procedure does not appear in the Graphics Display window. Output from each line differs because the PRINT, SHOW, and TYPE primitives cause Logo to do different things. SHOW displays the list itself--including the brackets--followed by a carriage return which causes the cursor to drop down to the next line on the screen. TYPE displays the items in the list--without the brackets. Notice the TYPE primitive also does not include a carriage return. That is why the output from the PRINT primitive is displayed on the same line. PRINT displays the elements of the list just like TYPE but it includes a carriage return.

Logo allows for much more sophisticated list handling. For example, in Digital Research's version of Logo for the ST, the instructions:

```
FIRST [SHERYN JOHNSTON]
LAST [SHERYN JOHNSTON]
```

allow you to output only one part of the input object. The input object in such an instruction is the word, or words, inside the brackets. In this example, SHERYN JOHNSTON is the input object. Output from the first line in this example would be SHERYN since that word is the first element in the input object. Output from the second line would be JOHNSTON since that word is the last element in the "input object."

Another primitive, BUTFIRST, is used in a similar way. Look at this example:

```
BUTFIRST [X 2 3 4 5]
```

Output from this instruction would be [2 3 4 5] since the BUTFIRST primitive outputs all elements of the list except the first one.

Sources of Additional Information on Logo

Although you could spend many hours experimenting with the primitives which have been introduced, this is by no means an exhaustive discussion of the available Logo primitives available in the ST version. Like the BASIC chapter, this chapter was not intended to make you an expert programmer. The following sources may be helpful sources of information if you are interested in learning more about Logo.

Introduction to ST Logo (MichTron, $49.95) is an easy-to-use tutorial on disk which could be helpful in offering more practice learning turtle graphics.

Numerous books and articles in magazines such as *Compute!*, *Antic*, and Analog--concerning ST Logo programming procedures are already available. In addition, at least fifty books have been written on the educational applications of Logo, and most of the educational computing journals regularly publish articles on the language. A recent double issue of *Computer's in the Schools* (Summer/Fall, 1985), for example, is a valuable source of information on Logo.

Several national organizations have also been formed to support Logo:

Young People's Logo Association publishes a monthly magazine, Turtle News, in several languages as well as books on Logo. In addition, the organization supports a computer bulletin board that distributes public domain Logo software and serves as a forum for discussing Logo topics.

National Logo Exchange also publishes a monthly newsletter for teachers using Logo in their classrooms. The organization also serves as a clearing house for reports and research on the use of Logo and as a conduit for the exchange of ideas and techniques for teaching Logo.

FOLLK (Friends of LISP/Logo and Kids) is an organization "dedicated to bringing 'languages for learning' along with other powerful ideas developed in Artificial Intelligence research, to folk of all ages, interests and levels of computer expertise." FOLLK serves as an educational resource and as an information network for sharing insights, ideas, and experiences among members. It publishes a quarterly magazine, FOLLK-Lore, and supports a computer bulletin board called FOLLK-Net.

13

Machine and Assembly Language

As we mentioned in Chapter 1, a computer actually understands only one language, machine language. However, it is not a single language. Each computer chip has its own set of instructions built into it by the designers. Instructions written in any other language must be translated into the machine language used by the computer's microprocessor chip before it can understand and process the instructions.

INSTRUCTION SETS

Just what is an instruction set? And how do computers use it? Inside the tiny microprocessor chip that is the heart of any personal computer is a maze of electronic connections that were formed when the chip was made. The pins on the outside of the chip's case are the only link between the chip and the rest of the computer. When the power is on, the chip accepts instructions on the input pins.

Each instruction a chip understands has a name and a number. The name is for humans; the number is for the computer. If, for example, the hexadecimal numbers 20 0A are sent to the 68000 chip in the ST, the chip interprets it as an instruction to load a section of the chip called the D1 data register with a number stored in one of the computer's memory cells. (Yes, 0A is a number. It is explained later.) This is a 4-byte instruction because it requires two bytes for the instruction and two more bytes for some additional information--in this case the memory location of the number that is to be loaded into the accumulator, a temporary storage bin on the chip.

The Motorola 68000 CPU is one of the more powerful chips currently in use. It is an improved version of the 6800 chip that was used in several personal computers a few years ago. The 6800 was an adaptation of the 6502 chip used in computers such as the Atari 800 and the Apple II+. The 6800 was so similar, in fact, to the 6502 that the company that created the 6502 sued Motorola. They lost, but the two chips were very similar.

The instruction sets of most CPU chips have four different types of instructions. Some direct the processor to perform a particular arithmetic operation. For example, there are instructions in 68000 machine language that tell the CPU to add a number to the number in one of the temporary storage locations inside the CPU. Other instructions direct the processor to input or output data be-

tween the 68000 chip and other devices or memory locations in the computer system. Quite a few of the 68000 instructions are used to control the execution of the program. There are several types of Jump instructions, for example, that tell the chip to go to another area of the program and begin executing the instructions located there.

A fourth type of instruction tells the microprocessor to perform a logical operation. Logic operations require some comment. Thus far we have been cheating a little on the way the instructions are numbered. Perhaps you wondered how 0A could be considered a number? 0A is a valid number in the hexadecimal number system. Hexadecimal (hex) numbers are explained later in this chapter. They are actually quite easy for humans to understand--that is why they were invented in the first place. The computer, however, wants nothing to do with them. The 68000 microprocessor chip accepts only binary numbers, which are numbers made up of only ones and zeros. The decimal number 12 is OC in hexadecimal and 00001100 in binary.

You can learn these two new number systems and translate freely between the decimal numbers you are accustomed to using and hex or binary numbers. However, the computer always works with binary numbers which are, unfortunately, the most difficult of the three systems for humans to understand and read. Several of the instructions the 68000 chip understands apply "logical operations" to binary numbers.

LOGICAL OPERATORS

Mathematicians have developed a peculiar, but useful, set of rules or logic for working with binary numbers, and most microprocessor chips have some of these logic operations built into their instruction set. The 68000 can handle binary numbers that are 8-bits long (a byte), 16-bits-long (a word), or 32-bits long (a long word). The examples below use bytes. Two common logic operations are AND and OR. Suppose the two bytes below were stored in two memory locations in the ST:

10110100 first byte
00110011 second byte

If an instruction called for the computer to *AND* the bytes in these two memory locations, the result would be obtained by applying the set of rules given below to each pair of bits:

If both bits are 0, the result is 0.
If both bits are 1, the result is 1.
If one bit is 0 and the other is 1, the result is 0.

Using these rules the result of an AND of our two bytes would be 00110000. The only way to get a 1 with AND is for both the bits to be 1.

Now let's look at the rules for *OR*:

If both bits are 0, the result is 0.
If both bits are 1, the result is 1.
If one bit is 0 and the other is 1, the result is 1.
If the two bytes are *ORed* the answer would be 10110111.

With OR the only way not to get a 1 is for both bits to be 0.

An example of a logic instruction in the 68000 chip is 62. That instruction performs an OR operation between a byte of data in the storage location in the chip called the status register and the next byte of data in the program. The result of the OR operation replaces the byte in the status register.

Logic operations are often used in programs to mask or get rid of unwanted parts of a byte. Suppose, for example, that a particular program obtained input from a peripheral that sometimes placed a one in the eighth bit of each byte and sometimes put a zero there. If all the information required by the computer is in the first seven bits, it may be to our advantage to see that the eighth bit in every byte is always zero. This can easily be done by using AND.

Suppose we put the value 011111111 in a memory location. If each incoming byte from the peripheral is put in the accumulator and an AND operation is performed between the accumulator and the byte in the memory location where 011111111 is stored, the eighth bit of the resulting number will always be zero, while the other bits will not be changed. The pattern of seven ones and a zero that is stored in memory is not altered by the AND operation and can be used over and over to modify the incoming data byte.

ALTERNATIVE NUMBER SYSTEMS

Up to this point the discussion has, for the most part, sidestepped one unpleasant fact about machine and assembly language programming. Only the higher-level languages accept and use the decimal numbers we learned long ago. As mentioned before, computers cannot understand the familiar base-10 number system. Instead, they use a system called binary, or base-2. An understanding of both binary and one other number system, hexadecimal or base-16, is necessary if you are to do any serious programming in low-level languages.

Binary Numbers

Remember the machine language codes described earlier? All the numbers were written in hex or hexadecimal. Before the computer can work with those numbers they must be translated into binary numbers. Several binary numbers are listed below with their hexadecimal and decimal equivalents:

Binary	Hex	Decimal
00111010	3A	58
01000000	40	64
00000000	00	00
01000111	47	71

At the beginning it is important to point out the difference between a number and a numeral. Numbers are quantities, numerals are the symbols used to represent quantities. As can be seen above, binary numbers are made up of patterns of ones and zeros. That is handy for the computer since it can only store information in two states which are represented in the ST by a 5-volt signal being present at a particular location (which represents a one) or zero voltage present (representing a zero).

Hexadecimal Numbers

Working with binary numbers is a frustrating task for most people. On early personal computers such as the IMSAI 8080, there were rows of toggle switches you used to input binary numbers into the computer. To input the decimal number three, it was necessary to set 8 switches to represent the binary pattern of 00000011. You flipped a switch up to represent a 1 and down to represent a 0. Hexadecimal numbers were then used in an effort to reduce the necessity of flipping switches thousands of times just to load a small computer program.

As the previous table shows, hexadecimal equivalents of binary numbers are much shorter. The table below lists the three commonly used number systems, their base, and the symbols or numerals available in each system.

Name	Base	Symbols Used
Binary	2	01
Decimal	10	0123456789
Hexadecimal	16	0123456789ABCDEF

Most of us learned our adding and subtracting in the decimal system. It uses ten different symbols and has a base of 10. If two numbers such as 6 and 7 are to be added, the resulting value cannot be expressed by any of the 10 numerals available. The decimal system uses the concept of place value to handle the problem. The symbols or numerals in the first or rightmost position of a whole number have values between 0 and 9. Numerals to the left of the rightmost digit have values that are 10 times the value they would have if they were moved one place to the right. Thus 6+7 is expressed in the base-10 or decimal system as "1 ten and 3 ones" or 13.

Binary-Decimal Conversion

If we consider only decimal and binary numbers for a moment, the table below shows the relationship between the two systems. In both decimal and binary, nothing is a zero. The first numeral in both systems is one, but the similarity stops there. 1+1 in decimal equals 2. In binary the addition of 1 and 1 is 10. In binary, 10 means "1 two and 0 ones."

The numerals in binary do not stand for the same numbers they do in the decimal system. It takes a four-digit binary number to represent the 16 values for 0 to 16 in the decimal system. Since the ST often stores data in memory as bytes that are 8-bits long, it would be interesting to see how big a decimal number can be expressed in a binary number that is eight digits (bits) long:

1	1	1	1	1	1	1	1	**Binary Number**
128	64	32	16	8	4	2	1	**Decimal Value**
255	127	63	31	15	7	3	1	**Decimal Total**

The decimal value of the binary number 10000000 is 128, and the decimal value of the binary number 11111111 is 255. Thus with an 8-bit binary number 256 decimal values can be expressed (0 to 255). Mathematically, 256 is "2 raised to the eight power" which is a shorthand way of saying you multiply 2 times 2 times 2 times 2 times 2 times 2 times 2 times 2. The result is 256. A 16-bit binary number can express 65,536 values, and since there are 16 address lines on older CPU chips like the 6502 and Z80, the value 65,536 is also the maximum number of memory locations that can be addressed or used by those chips under normal circumstances. The 68000 used in the Atari ST has 24 address lines. If you raise 2 to the

24th power you will get a number just over 16 million. With 24 address lines the 68000 chip can deal with over 16 million bytes of RAM and ROM! The design of the ST does not implement all those address lines and you are practically limited to around 4 million bytes, but that is still more than five times the normal maximum of computers like the IBM PC.

Binary Equivalents of Decimal Numbers

Decimal	Binary	Decimal	Binary
0	0	9	1001
1	1	10	1010
2	10	11	1011
3	11	12	1100
4	100	13	1101
5	101	14	1110
6	110	15	1111
7	111	16	10000
8	1000		

Hexadecimal-Decimal Conversion

Now what about hexadecimal numbers? Hex uses 16 different numerals. From 0 to 9 the numerals are the same in hex as in decimal. Once you pass 9 in decimal though, part of the value is carried over to the next column on the left. In hex there are six more numerals or symbols--A,B,C,D,E,F--that must be used before a *carry* operation is performed. In the decimal number system 9 plus 1 is 10, but in hex, 9 plus 1 is A. In hex the numeral D stands for the same quantity as the decimal numeral 13. The hex numeral 10--"1 sixteen and 0 ones"--equals the decimal value 16--one ten and 6 ones. The important thing to remember about hex numbers is that their place value is on a base of 16, not 10. Thus 2 in hex is 2 in decimal, and hex 20 is decimal 32--"2 sixteens." If you write programs in 68000 machine or assembly language, you will use hexadecimal numbers.

MACHINE LANGUAGE PROGRAMMING

A machine language program takes the relatively simple instructions understood by the computer and makes it do fancy things like keep track of your company's inventory or play a rousing Sousa march through the sound voice synthesizer. A segment of a machine language program is illustrated below:

Line	Address	Object Code	Source Code	Comment
1	00100A	D2 59	ADD.W (A1)+,D1	;SUM NUMBERS
2	00100C	53 42	SUB.W $1,D2	;DECREMENT CTR

Each of the columns in the program serves an important purpose. The first column is simply a set of line numbers that help you keep track of where you are in the program. That is not a problem with this two-line segment, but machine language programs often contain several thousand lines of code.

The second column lists the memory address where the instructions that follow are stored. The memory addresses are listed as hexadecimal numbers. The entry in the Address column of the first line tells you the first byte of machine language instructions on that line is stored in memory location 00100A. The second

byte of instructions is stored in memory location 00100B. Remember, however, that 00100A and 00100B are hexadecimal numbers.

The third column contains the operation codes or machine codes the computer understands--when they are converted to binary numbers. These are the numbers the computer uses to determine just what you want done. These codes are the machine language program. Everything else listed is simply helpful information the programmer needs to understand or modify the program.

The next two columns are most useful to the programmer. One lists mnemonics or source codes that represent the operation codes--usually called op codes. The source codes provide at least a hint of what the instruction does. The last column is for additional comments that will help the programmer understand what the line does.

The program segment above accomplishes relatively simple tasks. Let's look at each of the lines in detail.

```
00100A  D2 59  ADD.W  (A1) +,D1   ;SUM NUMBERS
```

The first line is an instruction to add two binary numbers. The machine language codes for the add operation to be performed is D2 59. The mnemonic for this operation is ADD.W. ADD tells you it is an addition operation and the .W is used to indicate the addition is to be performed on a "word" which is a 16-bit number. Other instructions add bytes (8-bit numbers) and long words (32-bit numbers). One of the numbers to be added is located in a memory location. Every memory location in the computer is assigned a number, or "address" and the address of the memory location for our number is stored in a temporary storage bin inside the 68000 chip named A1 or "Address Register Number 1." The parentheses around the A1 tells us the number in A1 is not to be added. That number is the address of the number we want to add. The second number is not stored in memory. It is in D1, another temporary storage bin or "register" inside the 68000 chip. (Instructions before this line in the program would have put the correct memory address in A1 and the proper number in D1.)

Now let's look at the second line.

```
2 0010C 53 42  SUB.W $1,D2  ;DECREMENT COUNTER
```

This line is another math operation. The machine language instruction 53 is stored in memory location 0010C and instruction 42 is in memory location 0010D. The 68000 machine language instruction 53 42 is one of several subtraction instructions. This one will automatically subtract 1 from the number stored in register D1 inside the 68000 chip. That register is sometimes used as a counter to keep track of how many times an operation is performed. For example, you might put the number $4F (the $ in front of the number usually means it is a hexadecimal number) in register D1 and then subtract $1 (hexadecimal 1 is the same value as decimal 1) each time a sum operation is performed. When the number in the D1 register equals 0, the program stops performing that operation and moves on to another task.

Programming in machine language brings you in direct contact with the computer. The instructions used are understood and acted upon by the computer without any translation, except converting numbers into binary. Machine language programs take far less memory than other types of programs and they are very fast.

The advantages of machine language are not enough, however, to entice many people into this form of programming. The programs are very long from a

programmer's point of view. Even something simple like adding two numbers and displaying the results on the screen takes several instructions.

Assembly Language

One of the most trying things about machine language programming is the fact the symbols used for the instructions are numbers, and they give us no indication of their meaning. Unless you memorize the numbers and their meaning, machine language programming can be a slow, boring, procedure that is punctuated by searches through the table of instruction codes.

Using assembly language to program is one way of avoiding the problem of instruction names that are not self-explanatory. Remember the column of source codes in the machine language segment discussed earlier? Source codes are short mnemonics that tell you what the instruction does. The mnemonic for a logical AND operation is AND. The mnemonic for an instruction that moves data from one of the address registers to another is MOVEA.

Assembly language uses the same syntax as machine language, and the vocabulary is equivalent. However, instead of a number, each instruction is a short mnemonic that provides a clue as to its function. In the example above, the machine language instruction for "add two 16-bit numbers" was D259, but the assembly language mnemonic for it was ADD.W. That is much easier to remember.

Some programmers write their program in assembly language and then make the conversion to machine language by hand once the program is written. That is not necessary on most computers. You can purchase a very useful program called an Assembler that lets you write programs in assembly language. Assemblers have commands that will convert or *assemble* the programs you write in assembly language source codes into machine language op codes the computer can process.

Simple assemblers do little more than convert your assembly language program into machine language the computer can run. Sophisticated assembler programs do much more. Many have a comprehensive set of editing commands you can use to edit and revise programs as you write. In addition, most have *macro* instructions that actually stand for a whole series of machine language instructions. When using an assembler with macros, it is possible to use one macro instruction that will be converted into many machine codes.

Some assemblers also come with a set of *subroutines*. Subroutines are much like macros. They are programs written to accomplish a particular task. When you need that task performed in a program, you can use the pre-written subroutine instead of writing the routine yourself. That saves time and reduces the likelihood of errors if the subroutines were written specifically for the computer you are using.

If you would like to explore machine and assembly language programming in more detail, there are several sources of additional information. One book you may find useful is *The Motorola MC68000 Microprocessor Family* by Thomas Harman and Barbara Lawson. It is published by Prentice Hall and provides a comprehensive introduction to the Motorola 68000 chip. The book covers both machine and assembly language, but the authors note that, "The extensive instruction set ... would preclude efficient coding in machine language except for the simplest of programs. In assembly language, instructions and addresses are designated by symbolic names which the assembler program translates into the appropriate binary code." Like Harman and Lawson we recommend you concentrate on assembly language programming instead of machine language if low-level languages are of interest.

Books like the one by Harman and Lawson only deal with the 68000 chip, there is no information on the way the chip is used in the ST or on the machine language subroutines in the ST's operating system. Many programmers take advantage of the subroutines in TOS when there is one that accomplishes what they want to do. They simply send the program to the section of ROM where the subroutine is located, run that part of TOS, and return to the next instruction in their program.

If you would like ST-specific information on 68000 machine and assembly language programming, books published by Compute! Publications should be helpful as well as several books in the ST series from Abacus. For more information, see the chapter on sources of information.

One other excellent source of information on machine and assembly language programming is a series of articles in *Antic* magazine. They are written by David and Sandy Small, two old time Atari devotees, and they deal with both the 68000 chip and writing programs on the ST. The first article in this series appeared in *Antic's* February, 1986 issue. *Antic* also published Chris Chabris' article, "Introducing 520ST Assembly Language" in the December, 1985 issue. *Compute!* magazine has also published a number of articles that will be of interest to the ST machine or assembly language programmer.

Assemblers for the ST

The ST is a relatively new computer, and there are not many assembler packages available for it at present. Atari's *ST Developers Kit*, which is used by professional programmers writing programs for the ST, has a relatively complete assembler in it. The Developers Kit includes an ST version of the C language, a 68000 assembler, several files of subroutines you can incorporate in your programs as needed, and a huge amount of information on the inner workings of the ST, GEM, and the operating system. The kit is available only from Atari (call 408 745-4926 and ask for Richard Frick), and it costs $300. This is not the place for novices to begin since it assumes you already know a great deal about programming.

A less expensive assembler is *A-SEKA*, a simple assembler package distributed by Antic Software for $34.95. Antic also has *GST ASM*, a more sophisticated assembler that supports macros and comes with a library of subroutines as well as programs that make it easy to create ST screen displays using the subroutines in GEM. It is $59.95. Antic Software's *GSTC Software Development Package* ($79.95) also has a 68000 assembler as well as the C language and several other programs that make it easier to develop ST programs in assembly language or C. The ST Software Catalog from Atari describes this program as "an ideal primer for the beginner, yet is powerful enough for the professional."

Another good assembler package is *DEVPAC ST*, a program that was developed in England by Hisoft. It is $79.95. Techstar of Scotts Valley, California also has an ST assembler, *Meta Assembler with Screen Editor*. It is $79.95.

If you become an accomplished assembly language programmer, you may also want to disassemble programs. A disassembler takes a machine language program and disassembles it into an assembly language program you can understand and modify. SCI Software Development sells an ST disassembler for $35.

Assembly language programming is fascinating to many computer owners. If you write your programs in assembly language you can deal directly with the special chips in the computer such as Yamaha sound synthesizer and the video controller chips. You will learn about the way these chips operate, and about how they are controlled in programs developed by commercial programmers.

Assembly language programmers also delve deeply into the "memory map" of the computer. They learn the location of hundreds of subroutines in the ST's operating system, the addresses of specific pins on the various chips in the system, and the memory locations that contain the data displayed on the screen. If you would like to explore the inner workings of the ST, we recommend assembly language programming. However, if you are just beginning your career as a programmer, we also recommend you start with a high-level language such as BASIC or Pascal before tackling assembly language. High-level languages do not place you in direct contact with the computer, but they are much easier to learn. We think they are a good beginning point even for the budding assembly language programmer.

14

Other High-Level Languages for the ST

Two high-level languages for the ST, BASIC and Logo, have already been discussed, and the previous chapter dealt with 68000 machine and assembly languages for the computer. Those languages will be enough for most people, but there are many other programming languages for the ST. In fact, there are indications the ST may become a preferred computer for hackers. There are ST versions of languages like C, Pascal, and Modula-2 that are very popular with hackers and full-time programmers. This chapter describes some of the other computer languages currently available for the ST and explains their positive and negative points.

OTHER BASICS

Most of the BASIC languages used by personal computers today were developed by a company called Microsoft. Microsoft BASICs are in Apple, Atari, Commodore, Radio Shack, and IBM computers. All of Microsoft's versions of BASIC are similar, but there are differences because the machines they run in are different. For example, you cannot take a program written in Apple's BASIC and run it on an IBM or Atari computer. That is why several versions of BASIC programs are often published in magazines such as *Compute!*. There may be one version for the Apple IIe, another for the IBM PC, and still another for the Atari ST or the Commodore Amiga.

Versions of Microsoft BASIC dominate the market today primarily because they are usually supplied with personal computers at no extra cost. Microsoft BASICs are not perfect, however, and they are not particularly suited to some types of programming. For example, the versions of BASIC that come with computers like the ST are interpreted languages. Programs written in a high-level language such as BASIC must be translated into machine language before they can be run by the computer. The program that does the translating takes each of the instructions written in the high-level language and converts it into machine language instructions. One BASIC statement may require a long series of machine language instructions to replace it when the translation is made.

BASIC Compilers and Interpreters

Translation is handled by two types of programs -- compilers and interpreters. A BASIC compiler takes a program written in BASIC and completely translates it into machine language. The program may then be run immediately or saved and run later. Once the compiler does the translation, the BASIC program need not be translated again.

An interpreter translates the program piece by piece and runs the program at the same time. That is, it translates enough to know what to do first, does that, then translates a bit more of the BASIC program, does that, and so on. At present almost all personal computers use interpreters rather than compilers for translation. Programs written in interpreted BASIC run much slower than compiled BASIC programs because interpreted programs must be translated into machine language each time they are run. Some business programs and programs that use animated color graphics run too slowly when written in interpreted BASIC.

Speed is the primary advantage of compiled BASIC. Ease of use is the primary appeal of interpreted BASIC. Errors are much easier to correct in interpreted BASIC. That is why personal computers come with an interpreted version of BASIC. *Henry's Fundamental BASIC* (available from Philon for $49) is an interpreted version of BASIC designed for student or home use. It comes with easy to understand documentation and is designed more for ease of learning than for writing commercial software.

If your programming work calls for a version of BASIC that is faster than the interpreted languages that come with the ST, there are several compiled versions of BASIC for the ST. *Softworks BASIC* ($79) is a compiled version of the language that has many advanced programming features not available in the regular interpreted BASIC supplied with the ST. It is available from *Softworks Limited.* Another compiled BASIC for the ST is available from Philon. *Philon Fast/BASIC-M* is aimed at the experienced programmer writing commercial software for the ST. The package is $129 and includes six disks as well as comprehensive documentation. Philon recommends you use a hard disk drive when writing programs in Fast/BASIC-M. *LDW BASIC ST* is another compiled version of the language that includes a number of ST-specific features. It is $119.95 from LDW Software.

Problems With BASIC

Although BASIC is by far the most popular programming language for personal computers, it has many critics who believe another language should be the first one most people learn. The criticisms of BASIC center on the fact the language allows people to write rather disorganized programs and still get them to work. Many computer scientists prefer languages that require programmers to be more structured and more systematic in the way they write programs.

Programmers often refer to programs written in BASIC by novices as "spaghetti code." That derisive term is a curt way of saying the program has been strung together with little regard for program organization. It may work, but subtle bugs in the program are difficult, often almost impossible, to find and correct.

Those criticisms are true of some programs written in BASIC, but it is not impossible to write organized, structured programs. Personally, we like BASIC. Our desks are a mess most of the time and our programs tend to develop and grow more like weeds than carefully planted rows of corn. Perhaps BASIC fits the less formal, less systematic way we do things.

PASCAL AND STRUCTURED PROGRAMMING

The next language to be discussed in detail is Pascal. It was developed in 1971 by Niklaus Wirth, a professor at the Technical University of Zurich in Switzerland. He named the language after Blaise Pascal, a 17th-century mathematician who developed one of the first mechanical calculating machines.

Like BASIC, Pascal was originally developed as a language for teaching students to program. That, however, is one of the few similarities between BASIC and Pascal. Kemeny and Kurtz developed BASIC so they could introduce students who would not be majoring in computer science to the benefits and advantages of computing. Wirtz developed Pascal so he could teach computer science majors how to write structured programs.

The structured approach generally has the greatest advantage when used by professional programmers to write programs that will be used by many people and revised, maintained, and supported by other programmers. Structured programming, and languages best suited to writing structured programs, have the disadvantage of being more difficult to learn.

Pascal is now a popular language in computer science classes, particularly in introductory programming courses. The language is also widely used in commercial programming environments today although its popularity is being threatened by newer languages such as Modula-2 and C. Many business applications programs in widespread use today were written in Pascal. The inherent structure and organization in a Pascal program, and the ability to break large programming jobs into many smaller modules, or blocks, that can be written by different programmers or different programming teams, makes languages like Pascal very attractive when major pieces of software, such as word processors, electronic spreadsheets or database management programs must be written. Long, complex programs written in Pascal are generally easier to read and understand. And since Pascal is a compiled language, they run much faster than would the same program in interpreted BASIC.

Available Pascal Versions

At least 75 versions of Pascal are available today, and at least 10 will run on the ST. There is also a Pascal standard, but few dialects follow the standard exactly. That means programs written in one version of Pascal are generally not compatible with other Pascal dialects.

Today many different versions of Pascal are in widespread use, but two have captured a significant percentage of the market. *Turbo Pascal*, which is distributed by Borland International dominates the market for four reasons. Versions of Turbo Pascal are available for many different computers including the Apple II+ and IIe (requires a special circuit card), CP/M computers such as the KayPro II machines, and the IBM PC and compatibles. Turbo Pascal is also relatively bug free. A number of other versions of Pascal had major bugs that made it difficult, if not impossible, to write serious programs that ran correctly. A third reason for Turbo Pascal's popularity is its price. While many other versions of the language cost hundreds of dollars, the retail price of Turbo Pascal is $69.95, and it is widely available from mail-order software companies for under $50.

Finally, the fourth reason for Turbo Pascal's popularity is the availability of accessory software from both Borland and third party suppliers. Borland, for example, sells a *Turbo Tutor* package ($34.95). Borland and other companies also sell libraries of subroutines (procedures you can incorporate into your Pascal programs) and accessory software that helps you create and edit programs in

Turbo Pascal. Borland's Turbo Pascal is thus one of the least expensive, best-supported versions of Pascal available today.

The CP/M version of Turbo Pascal will run on the ST if you have a CP/M emulation and a way of transferring the program from 5-1/4" disks to 3-1/2." We do not recommend that approach, however, because the CP/M version does not provide access to the desirable features of the ST such as a large memory, color graphics, and sound. We mention Turbo Pascal because it is the most popular version of the language today and because it is quite possible Borland will produce an ST version of the language in the near future.

The other "standard" Pascal is *UCSD Pascal*, so named because it was developed at the University of California at San Diego by Kenneth Bowles. A few years ago UCSD Pascal dominated the professional programming market because it was one of the first full implementations of the language that was available for many different computers. UCSD Pascal was licensed to software development houses by SofTech, a company that bought the commercial rights to the language from the Regents of the University of California. Unfortunately, SofTech's licensing program was costly and complicated, and when other good versions of Pascal became available many programmers quickly switched. SofTech went through some hard times and eventually decided to make UCSD Pascal easier to acquire and use.

TDI Software licensed UCSD Pascal from SofTech and now offers a version for the ST. If you are familiar with UCSD Pascal, TDI's version for the ST will be very familiar to you. The $79.95 package comes with three disks: a startup disk, a UCSD Pascal disk, and a "tool kit" disk that contains utility programs and subroutines.

TDI also has a Developer Version of UCSD Pascal that includes several additional programs and utilities that make it easier to develop commercial software for the ST. It is $149.95.

UCSD Pascal has several advantages and some disadvantages. The major advantage is portability. UCSD Pascal does not compile your program into machine language. Instead it compiles "p-code" which is an intermediate level between source code and object code. That p-code can then be run on many different machines. As a particular computer runs the p-code program, it makes the final translation into the machine language of the computer. That means a program written in p-code on one computer is relatively easy to run on another computer that has UCSD Pascal.

There is a price to pay for portability though. Compiling into p-code instead of machine language means you need a run-time module of UCSD Pascal to get the program to run, and the use of an intermediate step means p-code Pascal may not run as fast as other versions of the program written in another Pascal. UCSD Pascal is at its best in a programming environment where the goal is to produce a relatively complex piece of software that will be adapted to run on many different computers.

Virtually all versions of Pascal, including Turbo Pascal, are compiled. That means you must write your program using an editor and then use a compiler to convert the program to machine language code the computer can run. Using Pascal is thus a three-step process:

1) Writing the program with the editor.
2) Compiling the program.
3) Running the compiled version.

This three-step process is both tedious and difficult in some versions of Pascal. That is not the case with Turbo Pascal and several of the ST versions of

Pascal. These versions make it very easy to shift from the editor to the compiler. Inexpensive, easy-to-use versions of Pascal have done their part to increase the use of the language, but there is another development that may give Pascal the momentum needed to replace BASIC as the most popular language for personal computers, especially in high school programming classes.

In 1984, after several years of work by a select committee of high school educators and university professors of computer science, The College Board established an Advanced Placement test in computer science. Students attending participating high schools complete a one-year high school computer science course and then take the AP exam in computer science. If they pass the test, many colleges give students six semester hours of credit in computer science and permit them to begin their computer science coursework with one of the intermediate programming classes. High scores on the exam are also likely to give high school students hoping to major in computer science a competitive advantage. The AP exam in computer science is thus very important to many students, particularly those who plan to major in computer science in college.

The three-hour exam has two parts: a series of multiple choice questions and a situation or problem students must solve by writing a computer program in Pascal. Until recently well over 90% of the high school programming courses were taught BASIC. The rest used a variety of other languages including Logo, FORTRAN, COBOL, and Pascal. The Advanced Placement computer science exam requires students to write a program in Pascal, however. No other languages are acceptable, and the high school course to prepare students for the exam must focus on structured programming. Soon after the AP exam was announced, many high schools either added a Pascal programming class to their computer science curriculum or replaced a BASIC course with one on Pascal.

Programming in Pascal

Pascal is more difficult to introduce briefly and explain than languages such as BASIC or Logo, because the structure of the language, which supports structured programming, is more complex and more central to understanding Pascal than it is to understanding BASIC or Logo.

In BASIC it is relatively easy to begin working with the language. You could, for example, type the instruction:

```
PRINT 6502 + 68000
```

and when you press the RETURN key BASIC will begin to execute the instruction immediately. With few preliminaries, BASIC can be put to work performing simple tasks that illustrate how the language works. With Pascal, things are not so simple. Before the language will perform even a simple task like adding two numbers, you must provide much more information. There must be a line that names the program:

```
PROGRAM ADDNUMBERS
```

Another line is needed to tell Pascal where the program begins:

```
BEGIN
```

Then the program, which is only one line in this simple example, can be typed:

```
WRITELN (6502 + 68000)
```

Finally, Pascal must be told where the program ends

```
END.
```

The one-line program in BASIC becomes a four-line program in Pascal.

```
PROGRAM ADDNUMBERS
BEGIN
WRITELN (6502 + 68000)
END.
```

Even with the four lines entered into the computer, you cannot simply type RUN and watch the program execute. Pascal programs are written and edited with an editor. Once written, you compile the program, and then run it. If there are errors in your program, you must go back to the editor, revise the program, recompile it, and then run it again to see if the error was corrected.

Unlike BASIC, Pascal is not a language that lends itself to trial-and-error programming. It does, however, have many advantages over BASIC when you are writing long programs. BASIC programs can be written quickly, but their organization can make it difficult to read and understand what each section of the program does. Even the programmer who originally wrote the program may have trouble understanding what the various components in a long program do. Pascal programs, by contrast, are easier to understand, and thus easier to modify and support.

One of the reasons Pascal programs are easier to understand is that they do not use line numbers. You cannot tell the program to jump to a particular line with an instruction like GOTO 260. You must, instead, divide the program into blocks of code that do a particular job. Then you send the program to a particular block by giving it the name of that block. Generally the blocks in a program are assigned names that tell you something about what that block does. A block named DICE-TOSS, for example, might be the one that simulates the roll of dice in a Blackjack program.

Some Pascal Fundamentals

Although the actual instructions are not the same in different Pascal programs, the structure and organization of the programs are often quite similar. Some of the required, and commonly used, elements in a Pascal program are discussed in this section.

Program Name. As mentioned earlier, each program begins with a name. Programmers generally select names that describe what the program does. The format for naming a program involves typing the word PROGRAM at the beginning of the first line followed by the name selected with a space between the keyword PROGRAM and the name.

BEGIN and END. In a very short Pascal program with only one section of instructions, that section must start with the keyword BEGIN. After all the lines of instruction have been typed, you must type END. In programs with several blocks or subroutines of instruction, each block must use the keywords BEGIN and END to tell the program where the lines of instruction begin and end in the block. END thus comes after the last line of instructions in each block. To mark the absolute end of the whole program, type END with a period after it.

Assignment Statements. As with most languages, Pascal allows variable names such as A or NUM. In BASIC you might set A to equal 6 with the instruction LET

A=6. That instruction could be used anywhere in the program. In Pascal, you must declare the names of variables you will be using at the beginning of the program, between the program name and BEGIN.

At that point Pascal lets you write *assignment statements*, which are lines of information on the variables you will use in the program. You do not have to tell the program what value each variable will take, but you must list the name of the variable and identify its type. Variables that represent numbers can be integer (whole numbers), or real. Real numbers are sometimes called floating point numbers. Integers are numbers like 45, -678, and 666694. Floating point numbers such as 8.2, -965.330, and 78.0001, have decimals in them.

As mentioned earlier, variables representing text, such as "John Smith" are called string variables. Assignment statements are used to tell the program the names and type of variables you will be using. They must also tell the program about any constants. Constants are assigned a value at the beginning of the program and keep that value permanently. If you try to change a variable's type--such as from integer to real--or change the value of a constant, the program will treat that as an error.

Here is how a Pascal program might deal with several variables and two constants:

```
PROGRAM Amortization;
CONST  points = 3.5;
  loanfee = 45;
VAR  term, counter:INTEGER;
  rate, payment, total:REAL;
  name, address:STRING
BEGIN
program statements would appear here
END.
```

In the program segment above, the actual value of two constants, points and loanfee, are assigned, and the program is told three integer variables (term, points, and counter), three real variables (rate, payment, and total), and two string variables (name and address). Commas separate the names of variables and constants on the same line, and a colon separates the names of variables from their type.

Program Control. Many programs involve conditional instructions. They are instructions the program is to carry out only when certain conditions are met. Pascal uses WHILE DO statements to accomplish that task. The format for WHILE DO statements is:

WHILE (a condition is specified) DO (one or more instructions)

Procedures. Procedures are considered by many to be the most important aspect of programming in Pascal. In Pascal a procedure is a block of program code that follows the same organizational structure as the program itself. It begins with a procedure name (.e.g., PROCEDURE compute interest), has assignment statements for any constants and variables the procedure will use, and the BEGIN and END keywords bracket the actual lines of instruction in the procedure.

With procedures Pascal programs can be modularized. The main program block may consist of little more than an ordered list of procedure names. When the program comes to the name of a procedure, it finds that block of the program, executes the instructions there, outputs any data the procedure produced to the ap-

propriate destination (e.g., the screen, a printer, or another procedure), and then returns to the main program to find the name of the next procedure to be executed.

In many ways writing a Pascal program is like writing the outline of a paper and then writing the paper itself. The outline, with the main headings and subheadings, is like the main segment of a Pascal program. Instead of headings and subheadings, you list the names of procedures. The procedures, which are written after you have carefully thought through the outline of the paper, are the completed sections of the paper that correspond to the headings and subheadings in the outline. For many, this structured, top-down approach is the most effective, most desirable way to write programs.

Pascal and the St

Several versions of Pascal are available for the ST in addition to the UCSD Pascal from TDI. Optimized Systems Software, the company that created the operating system and BASIC used in the 8-bit Atari computers, has Personal Pascal. The name may suggest this is a simple, easy-to-use version of the language. It is not. The $74.95 *Personal Pascal* is a full implementation of Pascal with good documentation. Professional programmers could use Personal Pascal to develop commercial software for the ST. The language comes with a library of subroutines called PASGEM that make it much easier to integrate the programs you write into the ST's operating system environment and to use subroutines in the ST's ROM. Frequently used icons such as windows, drop-down menus, and icons are easy to incorporate in Personal Pascal programs. The language comes with a well-written, understandable manual.

Pascal and Programming

Even this brief overview of the way Pascal programs are written shows how well the design of the language fits structured programming. You can write structured programs in current versions of BASIC, but there is little in most versions to encourage it. It can also be done with Logo, but Logo is not a general-purpose language with all the keywords and features needed to write efficiently many types of programs. If learning to write structured programs is your goal, Pascal may well be the best language for it today.

It would be uncharacteristic of the field, however, if the situation remained stable for very long. The Atari ST has stimulated a great deal of interest in the community of programmers and software development houses that create computer languages. In one short year after the computer was announced, owners can select from a wide range of high-level languages for the ST. Supporters of these languages frequently praise their advanced features or power at user group meetings. Some of the other high-level languages available for the ST are discussed in the remaining sections of this chapter.

MODULA-2

Niklaus Wirth, the person who created Pascal, has developed a newer structured language, Modula-2. Many people believe it keeps the best of Pascal while adding features not available in Pascal. Wirth's experience with Pascal convinced him the language could be improved if the concept of modular programming were strengthened and encouraged by the language, and if the syntax or rules for writing programs were made more sensible and straightforward. Modula-2 insists you develop programs in a modular fashion with each module accomplishing a par-

ticular job. Programs are constructed from groups of linked modules that supply data to each other under very structured, very organized conditions. If you like the structure and order of Pascal programs, you will probably like Modula-2 even more.

One of the most popular versions of Modula-2 on the ST is the implementation by TDI Software in Dallas, Texas. The company licensed Modula-2 from Professor Wirth and uses some of the code he wrote in the ST version of the language. A complete Modula-2 package from TDI costs only $79.95 which is relatively inexpensive compared to other versions of the language. For that you get a Modula-2 compiler, a program you can use to link modules of your program into one large program, and a set of utility programs and subroutines.

When a new computer becomes available, a few companies will rush into production with programs and languages that work on the machine but do not take advantage of the inherent strengths and features available on the new model. ST software rushed into production might not include ways of harnessing the color and high resolution graphics of the ST, and it might not take advantage of the desktop metaphor of GEM. "Quick" languages for the ST may not have any convenient way of using the subroutines built into the ST's operating system software.

TDI's Modula-2 is not a rushed job. It is a full implementation of Modula-2 as described in Wirth's book Programming in Modula-2. In addition, it uses the GEM environment for programming and provides ways for you to write programs that use GEM menus, the ST's mouse, and the icon oriented visual environment of the ST. The language also comes with a package of subroutines and sample programs, and there is a way to incorporate subroutine calls that tap the power of the subroutines in the computer's ROM-based operating system.

The language has some nice touches. The editor, for example, is very powerful and works more like a word processor than the clunky, hard-to-use editors supplied with many languages. One of the most surprising things about TDI's Modula-2 is the manual that comes with it. It is actually understandable! That is very unusual in this business. TDI's manual is well-organized, thorough, and useful to beginning Modula-2 users as well as experienced programmers. It is not, however, a tutorial on Modula-2. The manual deals with TDI's implementation of the language on the ST. Beginners with no background in Modula-2 should read a general book on Modula-2 for background. The manual also does not document the GEM subroutines the language can use. You must get that information from another source such as the series of books published by Abacus Software.

Another unusual feature of TDI Modula-2 is the way it deals with compiler errors. When you compile code in many languages, the compiler stops as soon as the compiler finds the first error. With this language all the errors are identified and marked. If you have three errors in a program, all three may be identified the first time you try to compile it. That saves time and reduces frustration. At $79.95 TDI's Modula-2 is a bargain.

C

Many professional programming shops have shifted from Pascal to a newer structured language, called C, that produces faster, more efficient machine language programs. There are several C compilers that run on the ST. In fact, not only are there C compilers for the ST, there are several packages of utility programs that make it easier to write C programs on the ST. Insoft, a software company in Brookline, Massachusetts, has a series of utility packages for the ST. *The Graphics Function Library* ($59), includes subroutines and procedures for draw-

ing many types of graphics including two and three dimensional curves. The company also has C subroutine libraries for math functions, and for searching and sorting data. These libraries are in the *C Toolbox* series and sell for $59.95 each. They work with the Alcyon C Compiler.

Another package of utility software of interest to the C programmer is *GSTC*, a GEM software development package from Antic Software. This package includes many different types of utility programs, including subroutines for creating graphics and for linking your program to the GEM desktop. GSTC also includes a complete C Compiler. It is $79.95. Antic Software also plans to distribute *Lattice C* as well as *Meta Pascal* and an assembler program called *Macro Assembler*.

Another program many C programmers use is *MINCE*, a text editor you can use to write programs in several languages including C. It is $175 from Mark of the Unicorn.

Another C compiler is *Megamax C* from Megamax in Richardson, Texas. This C compiler is $199.95 and comes with the compiler as well as several utility programs and a library of subroutines. *Fast/C* is another C compiler package for the ST. It is $149 from Philon.

FORTH

Another high-level language with strong support from some programmers is Forth. The language was developed several years ago and has been used for robotics control and similar applications. Writing programs in Forth is quite different from working in BASIC or even Logo. In many ways it is more like writing assembly language programs although the code does have more English-like words than 68000 Assembly Language. Programming in Forth centers around writing procedures called "words." When you write a procedure or word, that word is placed in the Forth dictionary. When the program calls for the use of that word, Forth looks up the definition in the dictionary and applies the definition to the task at hand.

In Forth the use of procedures (e.g., words) is taken much further than it is in Logo, and that makes the language both more powerful and more difficult to master. That generalization applies to two ST versions of Forth. The Dragon Group in Elkview, West Virginia sells *4xForth* for the ST. It is a highly recommended, full implementation of the language that costs $99. The Dragon Group's version of Forth already supports multitasking, which means with it you can have the ST run more than one program at once, and the company has plans to bring out several sophisticated products that expand the power of the ST even more.

Both 4xForth and the manual that comes with the language were designed for experienced programmers. Beginners will have to struggle to master it. It is based on the *Forth-83* standard for the language as is the *Holmes & Duckworth Forth* for the ST that is distributed by Mirage Concepts. It is $49.95.

An organization with the unlikely name of Mind Mine Atari Computer Center in Bellevue, Washington also has a version of Forth for the ST. *MT-Forth-83* is $49.95, and *MT-Forth-83 Level 2*, which is an extension and enhancement of the basic compiler system, is $59.95.

If you would like to explore Forth further, we recommend writing to the Forth Interest Group. This is a very active group of Forth devotees who keep up with developments involving Forth. FIG publishes a magazine, Forth Dimensions, has a library of public domain and commercial software for Forth users, and sells books on Forth to members. General books on Forth that we can recommend for novices include Leo Brodie's *Starting Forth* and *Thinking Forth* as well as *Mas-*

tering Forth by Anita Anderson and Martin Tracy, and *Learning Forth* by Margaret Armstrong. All these are available from FIG.

Other High-Level Languages

If you are an experienced programmer with years of experience on a large mainframe computer, you are probably familiar with languages like FORTRAN and COBOL. FORTRAN, which is short for FORmula TRANslator, has been around for many years. It is still widely used in university and research settings where mathematical calculations are a major way computing power is used.

FORTRAN programmers can feel right at home on the ST because there is a version of that language available for the computer. *Pro FORTRAN-77* is available from Prospero Software in London, England for $149.99. Prospero also has a *PRO Pascal* compiler package that is also $149.95.

FORTRAN 77 is also distributed in North America by TDI Software in Dallas. TDI offers a basic version of FORTRAN 77 that includes a complete UCSD Pascal compiler as well for $149.95, and there is a more complete version, *FORTRAN 77 (Developer Version)* that is available for $219.95.

Another old favorite among mainframe programmers is COBOL, which is short for COmmon Business Oriented Language. Many of the programs running on the computer down at the local bank are probably written in COBOL. BOS National in Dallas, Texas has a version of COBOL for the ST. *BOS/MicroCOBOL* is $550 and a set of programming utilities is another $550. The company also has other programs and programming utilities that support and enhance its version of COBOL.

15

Accessories and I/O for the ST

This chapter explains what many of the accessories for the Atari ST do, what they won't do, and points out some of the pitfalls and problems associated with buying accessories and software for this computer.

Several types of peripherals and accessories provide ways of communicating information to the computer and receiving information from the computer. There are many accessories for the ST, but a good buy is a product that is fairly priced and does what it is supposed to do. Locating good buys requires careful shopping.

When compared to other machines with comparable features, the ST is very well priced. The low price of the ST often means the buyer has something left over for additional equipment that enhances the power or versatility of the system. In fact, a sizable percentage of ST owners end up spending more money on accessories, such as a video monitor and disk drives, than they spent on the computer itself.

That pattern, spending more on accessories than on the basic machine, is difficult for many people to accept. We are, for example, accustomed to buying automobiles that cost thousands of dollars and selecting accessories for it that cost hundreds of dollars. A Toyota van may cost $14,999 and a stereo system for it may be $450. The Atari 520 ST, on the other hand, costs less than $500 while a hard disk drive for it may cost $700 to $900. If we applied that ratio of base price to accessory cost to the Toyota van and its radio, the radio would cost more than $28,000! The point is, the money you spend on the basic computer is often only a small percentage of the total cost of a computer system that can do the jobs you want it to do.

A few years ago several home computer manufacturers took advantage of the public's inexperience with buying computers and accessories. They lowered the price of their basic computers to the point where they lost money on each unit sold. Once you bought their computer, however, you were locked into that company for accessories, which were priced so high the company made a windfall profit on each item sold. Some companies even went as far as trying to design their computer so no other company could write software or build accessories for it.

Fortunately, American computer buyers were brighter than anticipated. The companies who championed those business practices are no longer in the home computer market! To their credit, Atari rarely did that. The ST has standard serial and parallel ports which means hundreds of companies make peripherals that

can be connected to the ST with standard cables available at many discount stores. Atari even built a MIDI interface into the computer so owners could use the ST with any of over a hundred MIDI-compatible music devices made by many different manufacturers.

For the most part, you have a competitive environment in which to shop for ST accessories. Few items are so unique that you have only one supply source. In addition, Atari appears to have chosen to compete in the marketplace by offering excellent prices on both the basic computer and the accessories that go with it. Atari's ST-compatible monitors, disk drives, and printer are priced well below similar products from other computer companies, and they are often priced low enough to compete against third party manufacturers.

This chapter is not, however, an advertisement for Atari. Many companies make products for the ST that sell at competitive prices, usually lower than the price of similar items from Atari. In addition, the products from third-party suppliers will often have added features not available on similar accessories from Atari. This chapter will discuss the types of accessories available for the ST and review some of the products available.

Where to Buy Accessories?

Where should an ST owner buy accessories? If you subscribe to the magazines *START*, *ST-Log*, or *Antic*, you will see ads for hundreds of mail-order companies with toll-free numbers. Call them up, give your Visa or Mastercard number, and they ship products to you by return mail. Prices for products through the mail are often 10% to 45% lower than the price of comparable products available in local stores. Should you buy locally or through the mail?

We buy about half our computer accessories and software through the mail and half from local stores. In our opinion, beginners should seriously consider buying even more from local stores for several reasons. First, a good store will have someone who can provide advice, suggestions, and hints as well as take your money. A good store, however, can sometimes be hard to find. Few computer or software stores carry any products for the ST, and of those that do, the selection is usually very limited. One stuffy computer store salesman haughtily told us to check the nearby toy stores for Atari hardware and software. He meant that as a rebuke, but when an ST has better graphics, cheaper peripherals, and does most of the same work as computers he was selling at three times the price, we weren't really that offended.

It is true, however, that software and accessories for the ST are more often found in discount department or electronics stores than computer stores, and that does pose a problem. You will be fortunate if the salesperson in the computer section of K-Mart, Target, Best, or Service Merchandise can do much more than point to the the rack with ST accessories and ask if you will pay with cash or credit card. Even with that type of store, however, it is usually easier to get help if a problem occurs. If the product doesn't work or the software doesn't load, the store will usually exchange it or refund the purchase price. A mail-order company will usually do the same thing but you must return the product first. It can take several weeks to straighten out a problem even if the mail order supplier is very cooperative.

There is one area, however, where mail-order suppliers generally have an advantage. Few local stores carry the variety of products available by mail. Several mail-order companies sell over 500 different products for the ST. Few retail stores carry that many. Sometimes the only source for a product you want will be a mail-order supplier.

There are hazards when buying products from both local stores and mail-order companies. Local stores often have salespeople who have little or no understanding of your computer. Instead of telling you they do not know the answer to a question, many of them will give you the answer they think is most likely to make the sale. That is because many salespeople work on commission. The more they sell, the more they make. If we go into a store where we're not sure of the salespeople, we ask a few questions we know the answers to. If the salesperson answers them accurately and honestly, even if the answer may reduce the likelihood of a sale, we can put more confidence in the answers that person gives to other questions. This technique has produced some astounding answers.

The people who answer the phone at a mail-order supplier can also provide inaccurate information, however. Perhaps the best advice is to be very careful about making buying decisions on the basis of what a salesperson, especially one working on commission, tells you. We know this is a sweeping generalization that does a disservice to many fine salespeople who do an excellent job. They are, however, a decided minority in this field. We feel a store that has such a staff is worth an extra 10% to 15%. Their knowledge and advice can save you time, frustration, and money. For example, many ST owners in our area generally shop at a Federated Electronics store because one of the salespeople there knows the ST well and is willing to spend time helping with a purchase.

A common problem with mail-order companies is shipping delays. Think twice about ordering something that is not in stock. Many companies will tell you they are expecting a big shipment on Friday and will be glad to ship yours that afternoon. Big shipments are often late in this industry because companies cannot keep up with demand. The same product you want may be the object of desire by thirty thousand other ST owners. That Friday shipment may not get there for five or six Fridays. In addition, even if the shipment does arrive, there may be so many back orders that your order cannot be filled from that shipment. It is not uncommon for mail-order suppliers to charge the cost of a product to your credit card the day you place the order and then take weeks, even months, to send the product to you.

Again, however, we are painting a bleak picture with a wide brush. There are honest, responsive mail-order suppliers who do everything they can to provide quick service. As with local suppliers, we feel that type of company is worth a little extra cost because it saves you plenty of frustration.

In the next section of this chapter we describe some of the products available for your ST.

EXTRA MEMORY

The ST comes with more memory than any computer in its price range. The lower-priced 520 ST has 512K of RAM installed on the computer's main circuit board, and the 1040 ST has 1040K of RAM. However, the 68000 CPU in the ST is capable of using several megabytes of memory, and several companies have already begun marketing memory expansion kits.

There are really two aspects to adding RAM to the ST. The design of the 68000 CPU allows it to handle several million bytes of RAM, but the design of the memory circuits in the ST limit it to 1040K. That means there is no simple way to expand RAM beyond 1040K. Unless Atari redesigns the main circuit board in the ST, adding additional memory beyond 1040K is a task best left to the experienced hacker. It is a complex task which involves changing some of the memory support

chips in the ST as well as rewiring and rerouting some of the circuits. That is no small undertaking.

On the other hand, if you have a 520 ST, the computer already has the ability to handle more memory. You can buy memory upgrade kits from several suppliers and, theoretically, "do-it-yourself."

Converting a 520 ST to a 1040 ST is not straightforward. You must open the case, unplug the keyboard from the main circuit board, remove two metal shields that keep Radio Frequency Interference from radiating out from the circuit board, and then do some surgical work on the circuit board that reroutes some of the data and addressing signals to different locations on the board. The process involves cutting some of the copper traces on the circuit board and soldering jumper wires to the traces.

This is not a good "first project" for someone who has no experience working on printed circuit boards. If you are determined to upgrade your 520 ST but are not an experienced hacker, we suggest you do one of two things. Find a computer store that will install the upgrade kit for a fee. A fee of $50 would not be unreasonable for this type of work.

An alternative is to find someone who has the technical expertise and experience to install the RAM upgrade kit and get them to walk you through the process. If there is an Atari or ST user group in your area, there are probably several members who have already installed RAM upgrades. User groups are excellent sources of information and help. Attending monthly meetings and supporting the club with dues is a small price to pay for the help, fellowship, and information you get from active participation in the club.

An even better alternative is to pay a company to do the whole thing for you. When we wrote this chapter Computer Support would supply the parts and do the upgrade for a very reasonable $175. Since Computer Support is an authorized Atari Repair Center, they are a safer bet than do-it-yourself options.

PRINTERS

Buying a printer for the ST is relatively straightforward since the ST can use standard printers. There are three major types of printers in widespread use today.

Dot Matrix Printers

Dot matrix printers create a character by printing a pattern of dots. Most produce characters by knocking an inked ribbon into the paper with a series of wires in the print head of the printer. Dot matrix printers are generally fast, from 80 characters per second (CPS) to over 200, and even the least expensive models produce readable print. The characters are made up of tiny dots of ink on the paper. This technology is capable of producing anything from barely legible characters to print that comes close to the quality of a standard office typewriter. Naturally, the better the quality, the higher the price of the printer.

Dot matrix printers that work with the ST are available for between $125 and $2000. Mediocre dot matrix print is often called draft quality while better, but not perfect, print is called correspondence quality, or near letter quality. The best quality of print is called letter quality, but few dot matrix printers can achieve letter quality output, even if they claim letter quality in their ads. Some dot matrix printers do have several print modes, such as draft, correspondence, and near letter quality(NLQ). Draft mode is fast but produces the lowest quality of print,

while near letter quality generally approaches the output of a good typewriter but is much slower.

The disadvantage of low-priced, dot matrix printers is often poor print quality. The counterbalancing advantages, when compared to daisy wheel printers, are a lower price, high speed, and the ability to print both graphics and several different types and styles of text. Many dot matrix printers can produce very good quality graphics, and most are capable of printing text in boldface, italics, expanded print (double-size letters), and more.

Solid Character Printers

Solid character printers come in several different types and styles, but most of them use a daisy wheel print mechanism. Different styles of type can be used by changing the print wheel (the daisy wheel). They produce solid characters that generally cannot be distinguished from the characters printed by an office typewriter.

The advantage of daisy wheel printers like the Diablo 630, and models from companies like Qume, C.Itoh, Cardco, Juki, and NEC is high print quality. The output looks very good, but there are several disadvantages. First, this type of printer is relatively slow when compared to dot matrix models. A speed of 55 CPS is fast for solid-character printers. Most work at 20 to 45 CPS, and many of the inexpensive, under $250 models, plod along at less than 12 CPS. Printing a 20 page report on a printer that works at 12 or 13 CPS takes a very long time.

A second disadvantage of solid character printers is their limited or nonexistent graphics features. With a compatible dot-matrix printer, you can produce "screen prints," printed versions of the graphics on the ST's display. You can also output graphics produced by programs like DEGAS. No daisy wheel printer can print the ST graphics, a serious limitation since the graphics are so good. A third disadvantage is price. You can generally buy more printer for the money if you select a dot matrix model instead of a solid character printer.

If there are so many disadvantages to solid character printers, is there any reason to consider purchasing one for use with the ST? We think one type of user may still want a daisy wheel printer. If you will be using the computer for business correspondence that must have the appearance and quality of output from an office typewriter, you may need a daisy wheel printer. Otherwise, we suggest you shop for a dot matrix printer.

Several companies, including Atari, make compatible dot matrix printers with near letter quality modes that produce print almost as good as from a typewriter. In fact, many people prefer output from a high quality dot matrix printer because they can use special printing features such as italics, boldfacing, and oversize characters for headings and subheadings. There are some fields, however, that still consider the office typewriter to be the standard of comparison.

Laser Printers

This new type of printer became available for personal computer use in 1984. Several companies, including Hewlett Packard, Ricoh, Canon, Apple, Cor-Data, and many others, make laser printers for use with personal computers. Laser printers cost between $2000 and $6000 in the summer of 1986. Even the least expensive laser printers offer unparalleled print quality. Laser printers use the same image generating technology as office copy machines, and they can output both high quality text and graphics that is well beyond letter quality. Laser output is compared to the output from typesetting machines, not typewriters.

Laser printers are rapidly becoming the foundation for a new use of computers--desktop publishing. Schools, organizations, and small businesses are buying their own computer and laser printer to create printed materials that, until now, were produced by expensive typesetters. In the summer of 1986 there was no desktop publishing software for the ST, but that should change quickly. However, the ST was designed to work with Epson dot-matrix printers, and a number of the laser printers currently available will emulate the Epson FX-80 or MX-80 models. That means you can use a word processing program like Final Word to create documents and then tell the word processor you are using an FX or MX-80. The word processor will send the output to what it thinks is an Epson dot-matrix printer. The laser printer then accepts that data and prints it as high-quality, "near typeset quality" print. Laser printers with Epson emulation modes can also print ST graphics since the system was designed to output graphics to Epson printers.

General purpose printers usually come with one of two different types of interfaces--serial or parallel. Serial printers accept data a bit at a time with each character code being made up of 7 or 8 bits. Parallel printers accept data 8 bits at a time over 8 separate data lines. The ST has both a serial and a parallel port built-in, and you will need nothing more than the correct cable to connect most printers with serial or parallel interfaces to the ST. Cables for the serial and parallel ports on the ST were discussed in Chapter 2.

Plotters

A plotter is a device that can be used to draw figures and graphs under the direction of the computer. Most plotters use special ink-filled pens to draw, and most are capable of creating multicolored charts, figures, and graphics on both paper and clear acetate sheets (for overhead projectors). A few years ago Atari sold an inexpensive plotter, the 1020, that used narrow, adding machine size paper. It was more a novel toy than a serious product, however, and some of the thousands built for Atari have been dumped on the market by companies that buy surplus electronics equipment. We have seen it advertised for as little as $30 in Atari magazines. The 1020 was designed to work with the 8-bit Atari machines and is not compatible with the ST.

Professional plotters from companies such as Houston Instruments, Strobe, Wanatabe, and Sharp generally have serial or parallel interfaces and are connected to the ST with a regular cable. For example, Epson's HI-80 plotter is a professional quality model that retails for $599. It has a standard parallel interface and a standard IBM PC parallel printer cable can be used to connect the plotter to the computer.

That takes care of the physical interface, but Epson does not supply ST compatible software for the plotter. It does have CP/M software, however, and it could be used with the ST if you have one of the CP/M emulator programs. Epson supplies the software on a 5-1/4" disk, however, which means you must have a way of transferring the program from a CP/M computer to the ST and then saving it on an ST 3-1/2 inch disk. For most people, that is more trouble that it is worth. Many will wait for ST compatible driver programs to become available and then buy a plotter that works with the software. Another option is to use a plotter than can be programmed from a computer language like BASIC. That too is more trouble than it is worth for most people.

A third option is to use ST graphics programs such as Easy Draw or Degas to create charts, graphics, and figures, and then output to a standard graphics printer. You can produce very good graphics on ordinary dot matrix printers, and with color printers such as the one from Okidata, you can produce many types of

colorful graphics on the ST and then output them to the printer that prints them in color.

Video Monitors

Monitors were discussed in detail in Chapter 2. At present, the only supplier of ST-compatible monitors is Atari. That may well change in the near future, but for the moment, you are limited to Atari monitors or regular color or black and white televisions.

Apple Compatibility

The dominant force in the home and educational computer markets has always been the various incarnations of the Apple II. There is a huge library of excellent software for the Apple II, and it has always been the dream of some Atari and Commodore owners to have the best of both worlds--to buy the less expensive but technologically superior Atari computer and run software designed for the Apple II+, IIe, or IIc on it.

That dream has been encouraged by rumors that some company somewhere was about to announce and begin shipping an inexpensive software or hardware product that would allow computers such as the Commodore 64 or the Atari 800 to run most programs written for the Apple II. Competing manufacturers have even hinted occasionally that they could, or would, produce an Apple-clone interface for their computers. Over the years we have worked with a few products that claimed to provide Apple compatibility for Atari 8-bit computers, but none, not one, has worked.

In 1985 and 1986 a company named Mimic Systems of Victoria, British Columbia, Canada regularly advertised a $599 device named The Spartan that, so the ads said, allowed the Commodore 64 to run Apple II+ software. After months of full-page color ads, the device was still not widely available when this chapter was written, and we wonder whether it ever will be.

By the summer of 1986 a few companies were advertising Apple emulation programs for the ST. The ads implied they were already available, but a price was not given. If you called the company, they said the product should be available "soon." We have heard that before, many times.

If the Apple II emulation product were available, should you buy it? Unless the product costs less than $150 and allows you to run most of the software created for the Apple II+, IIe, and IIc, we advise against it. In 1986 the Laser 128 computer, which emulates the Apple IIc, was featured in ads from Central Point Software for $395. That price included a keyboard unit with a built-in disk drive, serial and parallel ports, and connectors for both composite and RGB monitors. A Laser 128 for $395, or a similar product from Franklin Computer for slightly more, would be a better buy than an Apple emulator program for $300 that ties up your Atari ST running Apple software. There are better things to do with a ST than that.

Video Game Controllers

Educational and recreational applications of the ST often call for a means of input other than the keyboard. Many arcade games, for example, do not work well if you must press keys to control movement. For many years Atari has sold an inexpensive joystick that works on all its machines as well as computers and video games made by many other companies.

Our young video game evaluators, including Amy Willis, did not rate the inexpensive model highly, and some reported the Atari joystick was not very durable. We use a Pointmaster joystick from Discwasher. It is more expensive than the Atari joystick but was rated better. Even the Pointmaster is not arcade quality, however. If you are shopping for joysticks you may want to "test drive" several before buying. Joysticks vary considerably in their feel and fit, and a device that does not fit your playing style can cause fatigue and low scores.

Companies such as Baylis, Discwasher, Spectravision, Suncom, and Wico manufacture higher-priced but more adaptable models in a variety of formats and styles. Wico's *Super Three Way*, for example, is sturdily built, and comes with three different interchangeable handles. Wico, which also manufactures joysticks and trackballs for arcade games, has a comprehensive line of products that includes joysticks priced from $18 to $33. If you are shopping for a sturdy, inexpensive joystick, Unisource Electronics' *Super Stick* fits the bill at $9.95.

Trackballs

There are more versions of the traditional joystick than any other type, but at least a few models of several other game controllers are also available for the ST since it uses the same types as the Atari 8-bit computers. Both Wico and TG Products market a trackball game controller you can substitute for a joystick on the ST. Trackballs let you roll a large round ball set in the top of the controller to direct the action of a game. This procedure is more precise than many joysticks and can be performed more quickly. Trackballs are generally much more expensive than joysticks.

Modems

Modems were discussed in the chapter on telecommunications, but we would like to mention here that some of the older models of standard modems have been dumped on the market at unbelievably low prices. We have seen regular, 300 BPS modems on sale for as little as $30.

If you would like to explore telecommunications without spending a great deal on a modem, you might begin with one of the bargain modems advertised in Atari magazines. Keep in mind, however, that special modems made for the Atari 8-bit computers are not compatible with the ST since it uses standard modems that connect to the serial port of the ST. Several Atari modems for 8-bit machines have been dumped on the market at fire-sale prices, but they will not work with the ST.

The modem market is currently in a state of flux, with prices changing almost weekly. A few years ago, most modems purchased were 300 BPS units, but today 1200 BPS models are the standard, and 2400 BPS models are fast becoming popular as their prices drop below $300. The prices of modems are quite fluid at the moment, and if you do not shop carefully, it is easy to pay much more than necessary.

For example, we saw an ad for an Avatex 1200 BPS modem that is Hayes-compatible (e.g., it understands the commands used by Hayes modems) for $239. The price was quoted in an ad for ST compatible products such as Atari monitors, printers such as the Star SG-10, and ST software. The Avatex modem is a general purpose modem, however, and it has been advertised in several other magazines for $89. We emphasize that the $239 modem and the $89 modem are the same modem. Shop carefully and widely if you are looking for a modem at a good price.

Mass Storage Options

Atari's SF354 and SF314 floppy disk drives have already been discussed in earlier chapters. Atari has also begun shipping a 10-megabyte hard disk drive, and the company has announced plans for several larger capacity hard disk drives.

The Atari floppy disk drives are fast, dependable drives. It is unlikely that someone will manufacture compatible drives with additional features that appeal to ST owners. The only way third-party suppliers are going to get a share of the ST disk drive market is to offer drives at a much lower price. That is just what one company, Shanner International, did. The Shanner SD-2000 is a disk drive system for the ST that includes two high density drives. The system is much smaller than two Atari drives and is advertised for $429.95. Since the retail price of two Atari SF314 drives is just under $600, the Shanner drives cost less but give you the same storage capacity.

Things are not always as they seem when it comes to third-party products. The price of the SD-2000 does not include shipping and handling, so add $15. That brings it to $450. The price comparison in Shanner's ad used the retail price of the SF314 and the mail-order price of the SD-2000. In the same magazine that carried the Shanner ad, SF314 drives were advertised for $200 each. If you add a more reasonable $10 for shipping, that brings the mail-order price of two SF314 drives to $410, which is $40 less than the cost of the Shanner drive system.

In addition, the Shanner system has two drives. You cannot buy one drive and then add another later. We would guess, however, that most people buy at least one Atari disk drive when they buy the keyboard unit. Since the ST can only use two regular disk drives, most owners would end up with an extra drive if they bought the Shanner drive system.

The 3-1/2" drives used by the ST are relatively standard units, and several companies are likely to begin manufacturing ST compatible drives. If Atari-compatible drives are available from mail-order companies for $200, we would not buy a compatible drive unless it cost us $160 or less.

Using IBM-Compatible 5-1/4 Inch Drives

David Small has been writing about Atari computers for several years. His early articles, written with his wife Sandy, were published in the now defunct *Creative Computing*. Currently he writes articles for *Antic* and *START*. In the November, 1985 issue of *Antic* he provided details on connecting a standard, IBM-compatible 5-1/4" disk drive to the ST.

PC-compatible 80-track drives are available from mail-order suppliers for less than $150, but this is not a project to be pursued lightly. It requires quite a bit of expertise, and some knowledge of how the PC drive works since you must modify the drives termination circuitry so it doesn't interfere with the operation of the ST drive. Perhaps an enterprising hacker will put together a PC-compatible disk drive with an ST cable on it and offer it for sale through the ST-specific magazines. Many people who own both an ST and an IBM PC (or use one at work) would be interested in such a drive.

A Simpler Way of Achieving IBM to ST File Compatibility

If tearing into your Atari disk drive and the innards of an IBM PC drive are a bit much, there is a much easier way to exchange disk files between an Atari ST and an IBM PC. A program from MichTron called *DFT* (short for Direct File

Transfer) will do the trick nicely. All you have to do is connect a regular serial cable with a null modem adapted between the serial port on the ST and the serial port on the IBM PC or compatible. Then you load the DFT program into the ST and into the IBM PC (two disks are supplied, one for the ST and one for the PC). Then you follow the directions.

With DFT files can be conveniently and easily transferred between the two computers. The DFT manual even includes a diagram for building your own null modem cable. The manual says "you'll need up to 125 feet of eight-conductor cable" but that does not mean you should drop by Radio Shack and buy that much cable. You can transfer data between cables located up to 125 feet apart, but if your computers are ten feet away in the same room, a 15-foot cable should do nicely.

DFT is a nice program that does more than most file transfer software. For example, type a simple command and you can view a directory of the files on the PC disk on your ST monitor. DFT is $49.95.

Hard Disk Drives

As was discussed in Chapter 2, at least three companies manufacture hard disk drives for the ST. The models available in the summer of 1986 were all 10 megabyte models, but larger capacity drives should be available by the time you read this. Supra Corporation's 20-megabyte drives were advertised at $1095 in the summer of 1986, and a 60-megabyte model for the ST was $1995. These prices are about twice the price of similar drives for an IBM PC.

Since the ST has much of the circuitry needed to use a hard drive built into the computer, many companies will probably be tempted to build drives for the ST. By the time you read this, hard drives should be available from many different companies, and in that competitive environment, don't be surprised to see hard drives available at prices well below the $600-plus cost of today's models.

Useful ST Software

Throughout this book we have discussed many different types of software, but the focus has generally been on applications programs. In this section we will discuss several utility programs for the ST. A utility program is one that does a necessary but routine job. There are utility programs, for example, that let you use part of the computer's memory as if it were another disk drive, and there are programs that let you copy software that is copy-protected. Some of more useful utility programs for the ST are discussed below.

Copy Programs

Many programs developed for the ST come on a copy-protected disk. That means you cannot make a backup of the disk. It also means you will be without the use of the program if something happens to the original disk. The hard shell 3-1/2" disks are relatively sturdy, but all disks are fragile. Dust, spilled coffee, an inquisitive cat, static electricity, and a thousand other things can turn the program disk into a useless piece of mylar.

You cannot make a backup of many ST programs using ordinary procedures, but current copyright laws expressly authorize making backup copies of software for archival purposes. Companies copy-protect programs to prevent someone from buying one copy and then making several copies to give or sell to others. However, copy protection probably does not stop the determined software pirate. It is a considerable inconvenience to legitimate owners who may lose the use of software they paid for because the disk is damaged.

To avoid that problem we use a program called *Copy II ST*. It will create backups of most copy-protected ST disks. Copy II ST is available in many computer and software stores or from Central Point Software in Portland, Oregon. The price is less than $39.95. Central Point Software is the best known company that makes copy programs, but several other companies also have similar programs. For example, Duplicating Technologies sells The ST Duplicator for $59.95 plus $3.50 for shipping and handling. The program will make copies of most ST programs.

MichTron Utilities

MichTron is a major supplier of software products for the ST, and *MichTron Utilities* ($60) is a collection of very useful programs. It includes a high-speed disk copy program that speeds up ordinary disk copying procedures and allows you to make backups of copy-protected software. The programs also let you explore and modify data in a disk file. You can change any byte of data in a file as well as the name of the file. You can even recover data from a disk that has been damaged.

MichTron Utilities is very similar to the Norton Utilities for the IBM PC. Both allow you to get behind the disk operating system and deal with the data on disks directly. That can come in very handy when you have problems and the normal operating system commands do not work. If you ever delete a file from a disk and then discover you deleted the wrong file, MichTron Utilities can be used to recover that file (if you have not stored additional files on the disk that used space occupied by the deleted file). A similar program, *H&D Tool Box Volume One*, is available from Holmes and Duckworth. The price is $40. Optimized Systems Software's *Personal Diskit* is also similar.

RAM Disk Programs

Another program from MichTron is *M-Disk*. It converts a portion of Random Access Memory in the ST to an electronic "disk drive." When you use M-Disk, it will reserve as much as 800K of memory (you decide how much) for a RAM-disk. The RAM-disk acts just like a real disk drive. You can store files on it and read those files. The two major differences between a RAM-disk and a real disk drive are speed and permanence. When you store or retrieve data from a RAM-disk, it works much faster than a physical drive. That makes RAM-disks handy when you are using programs that frequently store and retrieve data in disk files. Data stored on a RAM-disk is lost when you switch off the computer, so you must be sure to save the data on the RAM-drive to a real disk before switching off the computer. MichTron's *M-Disk* is $39.95.

Hippopotomus Software's *Hippo Ramdisk* ($34.95) is another RAM-disk program for the ST. MichTron also has other utility software such as *Soft Spool* ($39.95), a program that manages output to a printer. With Soft Spool you can send data to the printer. Soft Spool handles the transfer of data to the printer while you use the computer for other applications. You don't have to wait until all the data is transferred to the printer.

Other Accessories for the ST

Each month it seems several more companies offer interesting ST-compatible products. Miller Computer Products, for example, manufactures the *520 Station*. It is a nice desktop hutch that lets you arrange the ST and its peripherals into a nice, organized workstation. It has shielding that protects the disk drives

from magnetic fields, and the design of the hutch reduces the amount of cable clutter you find in most ST installations. The monitor sits on top of the hutch and two disk drives fit on shelves in it. The *520 Station* is available from Xanth Computer Systems for $99.95. Xanth also sells ST software and memory upgrade kits for the 520 ST.

Another interesting product that is particularly appealing to ST owners is a system for creating T-shirt iron-on graphics. Several companies make special ribbons for popular printers from Epson, Star Gemini, and Okidata. These ribbons use a waxy ink that will melt off the paper and onto a T-shirt when you apply heat. Some companies even sell special ink-pens you can use to add color to the printed graphic. With a special ribbon in your printer, you can output any graphic created by the ST to the printer. Print the graphic on plain white paper with the waxy-ink ribbon, and you have a T-shirt iron-on.

The only catch to this process is the necessity of creating a mirror image of the graphic. Since it is transferred to the T-shirt as a reverse image, you must be able to produce a mirror image of the graphic before printing it. That can be accomplished with several of the graphics programs for the ST. You can get heat transfer ribbons from several companies including American Ink Products, Diversions, and Ribbon Land.

Use T-shirts with at least some polyester or another synthetic fiber in them since the ink does not transfer well to 100% cotton shirts. Put something like aluminum foil inside the T-Shirt to keep the ink from bleeding through to the other side. Do not apply heat to the transfer as you would normally iron a shirt. That is, don't slide the iron across the surface of the transfer to melt the ink. Normal ironing procedures may cause the iron to smear. Instead, place the iron directly on one section of the transfer, let it stay there for a few seconds, then lift the iron up and place it on another section. Use a low temperature setting and avoid using travel irons that only have one heat setting--hot.

One good way to keep up with all the new products for the ST is to subscribe to one or more of the magazines that cater to owners of Atari computers.

16

Sources of Information and Software

Several million people in over twenty-five different countries use Atari computers, and a growing number of Atari enthusiasts are ST owners. The ST, in fact, is a very popular system in Europe, and several ST-oriented magazines are published in Germany and the Netherlands. The large and growing group of ST owners has attracted the attention of several publishing companies. At least 35 books about the ST have been published, and several excellent magazines publish at least three or four articles on the ST in each issue. A few magazines are devoted entirely to the ST.

BOOKS

Early in 1986 very few of the available books dealt with the ST, but that has changed quickly as the popularity of the computer has grown. Several introductory books are available that introduce the machine to the novice owner.

Presenting the Atari ST. This book is published by Abacus Software. It was one of the first books out on the ST and the authors clearly sacrificed quality in order to get the book out quickly. On many topics it covers less than the manuals that come with the ST. Few people will find this book useful.

The Elementary Atari ST. Written by William Sanders and published by Compute! Books, this is one of the better introductions. The title implies the book is a general introduction to the Atari ST. It is not. Most of the book is on programming in BASIC. There are chapters on Logo, Forth, and the GEM operating system, but six of twelve chapters cover programming in BASIC. If you are a novice and are interested in programming, this book would be a good place to start.

Atari ST User's Handbook. Written by Gilbert Held and published by Weber Systems Incorporated, this short book introduces the computer, explains how to use GEM, and covers the basics of Logo in one chapter. It covers much less than the book you are reading now.

The Atari ST User's Guide. John Heilborn's book is published by Osborne/McGraw Hill. It covers setting up the computer using standard Atari peripherals, using GEM, and programming in Logo. It provides more information on programming in Logo than the book you are reading and less on other ST topics.

Understanding the Atari ST. Tim Knight authored this book which is published by Sybex. The title implies it is a general introduction to the Atari ST, but it is actually a book on programming in BASIC on the ST.
Compute!'s Kids and the Atari ST. Edward Carlson's series of computer books for kids has been very popular. One of his newest covers the Atari ST. It is published by Compute! Publications and covers topics such as using GEM and programming in BASIC.
Compute!'s ST Programmer's Guide. This comprehensive book is a reference guide, not an introductory tutorial, to writing programs in both Logo and BASIC on the ST. It is of interest primarily to the intermediate and advanced ST owner with an interest in serious programming.

The Abacus ST Series

One publisher, Abacus Software, has already announced a relatively complete line of books on the ST. Abacus has published many books on the Commodore 64 and Commodore 128, but in 1985 the company expanded its product line to include both books and software for the ST. There are seven books in the ST series in addition to the introductory book mentioned above. Each is a thick reference book that deals with a specific aspect of the ST.
ST Internals. This book deals with the special chips used in the computer and with the disk operating system.
ST Tricks and Tips. This book consists primarily of subroutines you can add to your own programs.
GEM Programmer's Reference. This book covers the GEM operating system from a programmers point of view. It shows you how to take advantage of subroutines and features built into the ST's operating system.
Graphics and Sound. This book explains in detail the way the special graphics and sound chips in the ST operate. The book also shows you how to write programs using graphics and sound in several languages including C, Logo, BASIC, and Modula-2.
Logo User's Guide. This is a programming tutorial and reference book on the ST's version of Logo.
BASIC Training Guide. This is a comprehensive introduction to the ST version of BASIC.
Peeks and Pokes. This is an advanced book on writing programs for the ST in languages such as BASIC, C, Logo, and 68000 machine language.

The Abacus books, which cost $19.95 each, are written in Germany and translated into English for distribution in North America. The books tend to be heavy on technical details and often assume you have a background in machine and assembly language programming as well as some understanding of the way integrated circuits work.
Introduction to Sound and Graphics on the Atari ST. This book, by Tim Knight, is published by Compute! Publications. It is one of the first of what will probably be a flood of specialized ST books. This one deals with the sound and graphics features of the ST. Compute! Publications has also announced another specialized ST book, Elementary ST BASIC, that covers programming in BASIC on the ST.
1001 Things to Do With Your Atari ST. This book, by Mark Sawusch and Dave Prochnow, is one of a series published by Tab. The ST book includes many ideas, suggestions, and programs you can use by themselves or incorporate in programs you are writing. Chapters cover programs in areas such as education, financial, games, and statistics.

Atari ST Graphics and Sound Programming. This is another Tab publication. Written by Henry Simpson, it covers the hardware and software aspects of sound and graphics on the ST.
The GEM Operating System Handbook. Although not written specifically for the ST, this book is a detailed guide to GEM, the operating system that is standard on the Atari ST and available as an optional operating system on other computers such as the IBM PC. The author is Dave Prochnow and it is published by Tab.
Mastering GEM. This is another book about the GEM operating system. It covers using GEM on the Atari ST and IBM PC computers. The authors are Phillip Balma and William Fitler. It is published by Sybex.
The Complete Atari ST Sourcebook. This book is part of a series published by Arrays, Inc. It introduces and explains the ST, but it is primarily a source of information about the software and hardware products available for the ST.

Written by James Trageser, this book, like others in the Arrays series, is both useful and misleading. It is nice to have information about so many products in one location, but it is also misleading because virtually any book that tries to be a comprehensive guide to products for a particular computer is out of date almost as soon as it is published. Even magazines, where articles and reviews are normally written three to five months before publication, are often behind current events. Books like *The Complete Atari ST Sourcebook* are useful, but they are usually not complete or up-to-date.

MAGAZINES

Magazines are also an excellent source of information for ST owners and there are several Atari specific-publications we can recommend:
ANTIC - The Atari Resource. This magazine has been around for a long time. It was one of the best Atari-specific publications when the old Atari 800 and 400 computers were hot items in the early 1980's. Current issues of the magazine still include articles on the older 8-bit Atari computers such as the XL and XE models. During most of 1985 and 1986 *ANTIC* straddled the fence between the 8-bit Atari computers and the 16-bit ST models. Each issue was devoted primarily to 8-bit products, but a special section, ST *ANTIC*: The ST Resource, included several articles on the ST.

For example, the March, 1986 issue of *ANTIC* included ST articles on the MIDI interface, upgrading the 512 ST to 1 megabyte of memory, the Haba Hippo C language, producing color graphics in Logo, and new produce previews. *ANTIC* also published a comprehensive guide to hardware and software products for the ST that is regularly updated, and it binds a catalog of software available from *ANTIC* into each issue. The catalog includes many excellent ST programs.
START, The ST Quarterly. In the July, 1986 issue of *ANTIC*, the publisher's column was titled "Best of Both Worlds." It explained the quandary faced by the magazine as the world of Atari computer-owners evolved into two distinct groups--those who own and use 8-bit machines such as the 130XE and those who own an Atari ST. A survey of the magazine's subscribers indicated there were strong and divergent feelings about just how to handle the two groups. As the publisher put it:

"After considerable discussion and thought, ... Antic Publishing must continue its tradition of support to the 8-bit Atari users, as well as the entire Atari community. However, the powerful new ST computers require longer and more complex pieces than *ANTIC* can accommodate. Clearly, ST owners need their own high-level magazine. A quarterly publication with ST disk seems like a realistic approach for us....But take all the ST material out of *ANTIC*? No, that's dumb. For

one thing, almost half our respondents are interested in both types of computers and want *ANTIC* to continue covering the entire Atari product line.... *ANTIC* magazine will devote a minimum of 60 to 70 percent of each issue's editorial space to material for 8-bit owners. The rest of the editorial pages will carry timely and useful ST material.... *START*, The ST Quarterly, a new title from *ANTIC* Publishing, appears June 1, 1986 and is totally dedicated to Atari ST computers."

The upshot of all this is that most, but not all, of *ANTIC* will be on 8-bit Atari systems. However, many of the introductory and novice-level articles on the ST will also be published in *ANTIC*. Intermediate and advanced level articles, and articles of interest primarily to experienced users of the ST, will be published in the new disk/magazine publication, *START*, which will be published four times a year.

To reduce the irritation of potential subscribers who now have to face the prospect of paying two subscriptions instead of one, *ANTIC* Publishing is offering a combination subscription that includes four issues of *START* (each of which includes a disk of programs) and 12 issues of *ANTIC* for $59.95.

ANALOG Computing and ST Log. Another old-timer when it comes to magazines about Atari computing is *ANALOG*. Like *ANTIC*, it is an Atari-specific magazine that has been around for many years. In 1985 it began publishing articles on the new Atari ST computer. The December, 1985 issue, for example, included an article on telecommunications with the ST and a review of an ST video game. Then, in the April, 1986 issue of *ANALOG*, a new section, ST-Log, premiered. That section was devoted to the ST computer. It included reviews of several commercial ST programs, articles on programming the ST in BASIC and C, and descriptions of new ST hardware and software products. The editorial in that issue made it clear *ANALOG* was having the same problems as *ANTIC*. "It's apparent that many 8-bit owners think we're deserting their machines in favor of the newer ST models.

"At the risk of being redundant here, we want to again assure that there will still be plenty in these pages for the traditional Atari owner....Last issue Mike DesChenes' editorial promised that, rather than cut out 8-bit coverage, we would increase pages to accommodate all Atari computers. As you'll notice this month, we've done just that, with 132 pages instead of our usual 112."

Thus far, *ANALOG* has included ST-Log in the regular issue of *ANALOG* as a separate section of around 40 pages. Subscriptions are $28 a year for 12 issues of the magazine. If you would like the magazine and an ST-compatible disk that contains all the type-in programs published in the magazine, the price is $84 a year for 12 issues with disk. Thus far *ANALOG* seems to be planning to keep ST-LOG a section in the general Atari magazine. If ST sales continue to increase, however, we would not be surprised to see ST-Log emerge from *ANALOG* as an independent magazine.

COMPUTE!. Until recently this was an outstanding reference magazine for Atari computer owners in general as well as for ST owners. Published monthly, it carries articles on Atari models as well as machines made by Apple, Commodore, and IBM. Most issues contain at least two or three programs that will run on the ST and numerous articles for novices as well as intermediate level users. *COMPUTE!* also has several regular columns on topics such as educational computing and programming. Subscriptions are $20 a year.

COMPUTE!'S Atari Disk & Magazine. This publication is an outgrowth of *COMPUTE!* which began publication in the Fall of 1986. As this magazine catches on, expect more and more of the ST-oriented articles to appear in it instead of *COMPUTE!*. Each of the bimonthly issues will contain many articles on the ST as well as programs developed specifically for the ST computer. Each issue will also come with a disk that contains the programs described in the magazine. To use

them, you need only place the disk in your drive and load the programs into your ST. The price for that convenience is high, however. Each issue of the disk and magazine combination will sell for $12.95 on newsstands. Subscriptions are a bit less expensive at $59.95 a year for six issues.

COMPUTE!'s Atari Disk and Magazine was one of the first ST-specific magazine/disk publication. It covers many topics including games and educational software, reviews of commercial ST products, and programming in languages such as Pascal, Logo, BASIC, and assembly language. The publication also includes a section on new products for the ST family of computers. It is expensive, but worth the price if you are a dedicated ST owner.

Atari Explorer. Published six times a year by Atari, this magazine publishes articles on all the Atari computers. Recent issues have included a special section of articles on the ST. The magazine covers all aspects of personal computing, from recreational uses to business applications and programming. Subscriptions are $15 a year for six issues. The magazine is well illustrated and generally has several reviews of commercial programs for the ST as well as in-depth articles on topics such as the 68000 microprocessor and programming in Logo.

PUBLIC DOMAIN SOFTWARE FOR ATARI COMPUTERS

We have already mentioned that information utilities such as CompuServe and Delphi have databases of information on the ST as well as software you can download and use on your computer. Public domain software that runs on the ST is also available from several other sources.

Public domain software, which can be freely copied and distributed without violating copyright laws, is available in reasonable quantity for the Atari ST. The number of programs available as well as their quality should increase rapidly as more people learn to write programs on the ST.

Several groups have collected a large number of programs for Atari computers and make them available for a small fee. You will generally pay a small handling and media fee, often $10 a disk or less. Each disk will contain from 10 to 40 programs.

Quality varies from terrible to great, but that is the same quality range you will find in commercial software. The major difference between public domain and commercial software is the documentation for public domain programs. Many programs simply don't have any. You have to figure out how the program works without it.

Other programs have documentation but it was often written quickly by the person who wrote the program. Without doubt, the worst person to write documentation for a program is the programmer who created it. It is very difficult for the creator to remember how it is to begin using the software with no idea of how it works. Many public domain programs require a bit of extra effort to figure out how they work.

User Groups

Several Atari user groups have large libraries of public domain software that is available free to members of the group. *Atari Explorer*, the magazine published by Atari, occasionally publishes a list of active Atari user groups, and the other Atari-specific and ST-specific magazines also occasionally provide information on active user groups.

Another national publication, *The Computer Shopper*, prints a list of several thousand user groups in each monthly issue. *The Computer Shopper* is a large, newspaper-format publication that has articles on many different types of computers, including the ST. The main appeal of *The Computer Shopper* to many people, however, is the advertisements of discount mail-order computer suppliers that sell everything from bulk packs of 3-1/2" disks to printers, all at very low prices. If you are looking for an Atari or ST user group in your area, you can probably find the address of a group in the current issue of *The Computer Shopper*. It is available on many newsstands.

Software from Antic

If you subscribe to CompuServe, you can type GO ANTIC and reach the Atari special interest group sponsored by *ANTIC* magazine. This SIG has files of information on the ST as well as many programs you can download. There are also several files of Atari graphics you can download and display on your monitor or output to a printer.

ANTIC magazine also distributes commercial software through a mail-order service. Catalogs are included in *ANTIC* and *START* magazines. *ANTIC* also distributes several hundred public domain programs at a nominal cost. Most of them are for 8-bit Atari machines, but the number of ST programs is increasing rapidly. For example, *ANTIC* distributes STTERM, a telecommunications program for the ST that lets you communicate with other computer systems. It has many features including the ability to upload and download programs using the Xmodem protocol, and it works at 300, 1200, and 9600 BPS. STTERM costs $12.

Programs from ANALOG

Not to be outdone, *ANALOG* magazine also sponsors an Atari special interest group. It is on Delphi, a popular and inexpensive information utility that was discussed in the chapter on telecommunications. You can reach *ANALOG*'s Atari group on Delphi by typing GR ATARI. There are a number of public domain programs for the ST in the program library of the group.

Commercial Distributors of Public Domain Programs

Several companies such as PC-SIG are in business to distribute some of the thousands of public domain and "shareware" programs available for popular computers such as the IBM PC and Apple II. Shareware programs, by the way, are programs that can be copied and distributed freely. The developer of a shareware program asks, however, that you voluntarily pay for the program if you find it useful. Several shareware programs, such as PC-Write for the IBM PC, are very popular today.

As far as we know, none of the large shareware/public domain software distributors has begun stocking programs for the ST. That means you will have to rely, for the moment, on user groups and Atari or ST Special Interest Groups on the computer bulletin boards and information utilities such as CompuServe and Delphi. We would be surprised, however, if at least a few distributors don't begin to collect all the public domain programs for the ST and begin offering disks of software for $10 to $15 a disk. Check the ads in magazines like *ANTIC* and *ANALOG* (or the ST spinoffs from those magazines) for potential suppliers.

Appendix: Sources of Software and Accessories

SOFTWARE AND ACCESSORIES

Abacus Software, 2201 Kalamazoo S.E., Grand Rapids, Michigan 49510 Phone: (616) 241-5510

Academy Software, Inc., P.O. Box 6277, San Rafael, California 94903 Phone: (415) 499-0850

Activision, P.O. Box 7286, Mt. View, California 94039 Phone: (415) 960-0410

Advanced Computer Systems, 8067 Janes Ave., No. B, Woodbridge, Illinois 60517 Phone (312) 357-1070

Alpha Systems, 4435 Maplepark Road, Stow, Ohio 44224 Phone: (216) 688-9014

American Ink Products, 527 Howard Street, San Francisco, California 94105 Phone (415) 982-0161

Antic Software, 524 Second Street, San Francisco, California 94107 Phone (415) 957-0886

Artworx, 150 N. Main Street, Fairport, New York 14450 Phone: (716) 425-2833

Atari Corporation, 1196 Borregas Avenue, Sunnyvale, California 94086 Phone: (408) 745-2000

Batteries Included, 30 Mural Street, Unit 9, Richmond Hill Ontario L4B 1B5, Canada Phone: (416) 881-9941

Bay View Software,177 Webster Street, Suite A-295, Monterey, California 93940 Phone: (408) 373-4011

Beckmeyer Development Tools, 592 Jean Street, No. 304, Oakland, California 94610 Phone: (415) 658-5318

Black Patch Systems, P.O. Box 501, Arnold, Maryland 21012 Phone: (800) 282-0402

Borland International, 4113 Scotts Valley Drive, Scotts Valley, California 95066

BOS National, Inc., 2607 Walnut Hill Lane, Suite 200, Dallas, Texas 75229 Phone: (214) 956-7722

Coincidence Software, 3714 - 140 S.E., Bellevue, Washington 98006 Phone: (206) 643-9695

Commnet Systems, 7348 Green Oak Terrace, Lanham, Maryland 20706 Phone: (301) 552-2517

Computer Place, 2160 W. 11th Avenue, Eugene, Oregon 97402 Phone: (800) 452-8013

Computer Support, 52 South Linden Avenue, #1, San Francisco, California 94080 Phone: (415) 589-9800

Digital Vision, Inc., 14 Oak Street, Suite 2, Needham, Massachusetts 02192 Phone: (617) 444-9040

Diversions, 1550 Winding Way, Belmont, California 94002 Phone (415) 591-0660

The Dragon Group, 148 Poca Fork Road, Elkview, West Virginia 25071 Phone: (304) 965-5517

Eclipse Engineering, P.O. Box 147, West Jornda, Utah 84084 Phone: (801) 964-1108

Duplicating Technologies, 99 Jericho Turnpike, Suite 302A, Jericho, New York 11753 Phone: (516) 333-5805

Electronic Arts, 2755 Campus Drive, San Mateo, California 94403 Phone: (800) 448-8822

Epyx, 1043 Kiel Court, Sunnyvale, California 94089 Phone: (408) 745-0700

Firebird Licenses, Inc., P.O. Box 49, Ramsey New Jersey 07446 Phone: (201) 9344-7373

Haba Systems, 6711 Valjean Avenue, Van Nuys, California 91406 Phone: (818) 901-8828

Hippopotamus Software, 985 University Avenue, Suite 12, Los Gatos, California 95030 Phone: (408) 395-3190

Hisoft, 180 High Street North, Dunstable, Beds LU6 1AT, England 0582-696421

Hybrid Arts, 11920 W. Olympic Blvd., Los Angeles, California 90064 Phone: (213) 826-3777

Infocom Software, 125 Cambridge Park Drive, Cambridge, Massachusetts 02140 Phone: (800) 262-6868

Insoft, Corp., P.O.Box 180, Boston, Massachusetts 02123 Phone: (617) 739-9012

Mark of the Unicorn, 222 Third Street, Cambridge, Massachusetts 02142 Phone: (617) 576-2760

MichTron, 576 South Telegraph, Pontiac, Michigan 48053 Phone (313) 334-5700

Microsoft Corporation, 16011 NE 36th Way, Box 97017, Redmond, Washington 98073-9717

Micro-W Distributing, Inc., 1342 B, Route 23, Butler, New Jersey 07405 Phone: (201) 838-9027

Migraph, Inc., 720 S. 333, Suite 201, Federal Way, Washington 98003 Phone: (206) 838-4677

Miller Computer Products, 14640 N.E. 32nd, Suite D4, Bellevue, Washington 98007-3649 Phone: (206) 624-9292

Mirage Concepts, Inc., 4055 W. Shaw, No. 108, Fresno, California 93711 Phone: (209) 227-8369

Oasis Systems, 7907 Ostrow Street, Suite F, San Diego, California 92111 Phone: (619) 453-5711

Optimized System Software, Inc., 1211-B Kentwood Avenue, San Jose, California 95129 Phone: (408) 446-3099

Oxxi, Inc., 3428 Falcon Avenue, Long Beach, California 92807 Phone: (213) 427-2080

Penguin Software, Inc./Polarware, 830 4th Avenue, Geneva, Illinois 60134 Phone: (312) 232-1984

Philon, Inc., 641 Avenue of the Americas, New York, New York 10011 Phone: (212) 807-0303

Power Systemss, 43328 S.E. 176th Street, North Bend, Washington 98045 Phone: (206) 888-2828

Progressive Computer Applications, 2002 McAuliffe Drive, Rockville, Maryland 20851 Phone: (301) 340-8398

Prospero Software, 190 Castelnau, London, England SW13 9DH Phone: 01-741-8531

Quantum Microsystems, Inc. P.O. Box 179, Liverpool, New York 13088 Phone: (315) 451-7747

Quickview Systems, 146 Main Street, Suite 404, Los Altos, California 94022 (415) 965-0327

Regent Software, 7131 Owensmouth, Suite 45A, Canoga Park, California 91303 Phone: (818) 883-0951

Ribbon Land, P.O. Box 8634, Red Bank, New Jersey Phone: (201) 842-1437

Royal Software, 2160 West 11th Avenue, Eugene, Oregon 97402 Phone: (503) 683-5361

SCI Software Development, P.O. Box 391807, Mt. View, California 94039 Phone: (415) 967-3288

Shanner International, 453 Ravendale Drive, Mountain View, California 94043 Phone: (800) 423-8673

SoftLogik, 429 Old Baumgartner, St. Louis, Missouri 63129 Phone: (314) 894-8608

Softworks Limited, 2944 N. Broadway, Chicago, Illinois 60657 Phone: (312) 975-4030

Spinnaker, One Kendall Square, Cambridge, Massachusetts 02139 Phone: (617) 494-1200

SST Systems, P.O. Box 2315, Titusville, Florida 32781 Phone: (305) 269-0063

SubLogic, 713 Edgebrook Drive, Champaign, Illinois 61820 Phone: (800) 637-4983

TDI Software, 10410 Markinson Road, Dallas, Texas 75238 Phone: (214) 340-4942

Techstar, Inc., 5353 Scotts Valley Drive, Suite E, Scotts Valley, California 95066 Phone: (408) 438-7201

Unison World, Inc., 2150 Shattuck Avenue, Suite 902, Berkeley, California 94704 Phone: (415) 848-6666

Versasoft Corp., 723 Seawood Way, San Jose, California 95120 Phone: (408) 268-6033

VIP Technologies, 132 Aero Camino, Santa Barbara, California 93117 Phone: (805) 968-4364

Xanth Computer Systems, 600 First Avenue, Seattle, Washington 98104 Phone: (800) 647-7741

XLent Software, P.O. Box 5528, Springfield, Virginia 22158 Phone: (703) 644-8881

LOGO INTEREST GROUPS

FOLLK (Friends of LISP/Logo and Kids), P.O. Box 22094, San Francisco, California 94122

National Logo Exchange, P.O. Box 5341, Charlottesville, Virginia 22905

Young People's Logo Association, 1208 Hillsdale, Drive, Richardson, Texas 75081

MAGAZINES

Analog Computing and *ST Log*
Analog Computing, P.O. Box 615, Holmes, Pennsylvania 19043

Antic and *START*
524 Second Street, San Francisco, California 94107

Atari Explorer
7 Hilltop Roade, Mendham, New Jersey 07945

The Computer Shopper
P.O. Box 1419, Titusville, Florida 32781-9988

COMPUTE!
COMPUTE! Publications, Inc., 825 7th Avenue, New York, New York 10019
Phone: (212) 265-8360

Index

Numbers

1st Word 27, 29, 48, 56, 67-8, 101
4xForth 205
8-bit forum 97
16-bit forum 97, 134
520 Station 217
1001 Things to Do with Your Atari ST 220
6502 chip 187
68000 chip 5, 16, 17, 187-88, 189, 191, 192, 194, 209

A

Abacus Software 219, 220
A-Calc 74
A Mind Forever Voyaging 113
A-SEKA 194
Academy Software 117
accessing mainframes 47, 85, 95-6
accessories 207-18
Activision 102, 112, 115, 134
activating windows 50, 140
adaptive user 7
Advanced Computer Systems 64
adventure simulations 107, 110-13
Alpha Products 18
Alternate (ALT) key 12
alternative number systems 189-91
Alycon C Compiler 205
Amiga 14, 16
Analog 91, 96, 137, 166, 222
Anchor Automation 84
Anderson, Anita 206
Animator 125
ANSI 88
Antic 15, 91, 137, 186, 194, 208, 215, 221, 224
ANTIC ONLINE 97
Antic Software 74, 128, 194, 205, 224
Apple compatibility 213
Appleworks 79
arcade games 115
arcade simulations 107, 113-15
area expert 7
arithmetic conditioners 155-6
Armstrong, Margaret 206
arrays 158-60
Art Gallery I 133
Artworx 107, 108, 109
ASCII 88
assemblers 193, 194, 205
assembly language 5, 187, 193-4, 194, 195, 205
assignment statements, Pascal 201-2
Atari Explorer 223
Atari Software 117
Atari ST Graphics and Sound Programming 221
Atari ST Owner's Guide 9, 25
Atari ST User's Guide 219
Atari ST User's Handbook 219
Atari TOS System Disk 24, 25, 26, 41, 51
Atari XE 104
Atari XL 104
Atascii 88
Audio Feedback 43
AUTO 148
auto-answer modem 84
auto-dial modem 84
automatic footnotes 67
automatic indexing 67
Avatex 84, 214

B

back-up disks 216
BACKWARD 171-2
Ballyhoo 113
Balma, Phillip 221
bank account management 105
banking services 94
BASIC 4, 5, 36, 96, 100, 137-67, 170, 195, 196-7, 200, 201, 203, 205
BASIC commands 146
BASIC editing 142-6
BASIC statements 146
BASIC Training Guide 220
BASIC windows 138
Batteries Included 71, 81, 123
baud 84
Bay View Software 107

binary-decimal conversion 190-1
binary numbers 188-91
bit 161
bit-mapped graphics 161
Black Patch Systems 16
block modes 66
board games 106, 107-8
Bonner, Paul 70
Borland International 198
Borrowed Time 112
BOS 81-2, 206
BOS/MicroCOBOL 206
Bowles, Kenneth 199
BOX 174, 177
BPS 84, 85, 86
Bridge 4.0 108
Brodie, Lee
bulletin boards 88, 90-1, 96
business applications 72-82
Business Operating System 81-2
Business Template 78
BUTFIRST 185
byte 1, 14, 15, 54
Byte 77

C

C 4, 194, 204-5
C Toolbox 205
cable clutter 10
cables, printer 19, 23
CAD-3D 99, 100, 128
Calendar 106
Cancel Box 44
Cards 110
Carlson, Edward 220
cartridge port 20-21
Casio 101, 102, 103, 135
CB 93
cell 72, 73
Central Point Software 217
central processing unit 5, 16, 17, 187, 209
Centronics 19, 47
Chabris, Chris 194
changing screen colors 44-45, 161-2
changing window size 52-3, 170
Check Minder 105
CIRCLE 162, 164-5, 174, 177, 179
CLEARSCREEN (CS) 173
CLEARTEXT (CT) 173
clock card 21
Close Box 49
closing windows 28, 49, 62, 141
CLOSEW 141
CLR/Home key 13
COBOL 96, 200, 206
Coincidence Technical Software 87
COLOR 161-2
command-driven word processors 65
COMDEX 17
Command window 139, 140, 141
Commnet Systems 87, 91
Commodore 2, 3, 4, 16, 74
comparing variables 155-6
compatibility, Apple 213
compatibility, CP/M 17
compatibility, Hayes 85
compatibility, IBM 215-16
compatibility, software 77
compilers 197, 199, 204, 205
Complete Atari ST Sourcebook, The 221
Compubridge 109-10
CompuServe 6, 81, 86, 89, 92, 93, 94, 96, 97, 98, 134, 224
Comp-U-Store 94
Compute! 137, 186, 194, 196, 222
Compute! Publications 194, 219, 220, 222-3
Computer Aided Instruction (CAI) 116
Computer Assisted Instruction (CAI) 116
Computer Aided Learning (CAL) 116
Computer Associates 11
computer bulletin boards 88, 90-1, 96
computer club 909
computer literacy 4
Computer Palace 10
Computer Shopper, The 91, 224
ComputerEyes 133
Computers in the Schools 186
Confirm Copies 32
Confirm Deletes 32
connecting monitors 13
Consumer Electronics Show 3
consumer services 88, 94
continuous feed paper 47
Control key 12
Control Panel 42-46, 161-2
Copy II ST 217
copy-protected disks 216
copying disks 56-8
copying files 58-9
CP/M 16-17, 93, 199, 212

CPU 5, 16, 17, 187, 209
creating folders 52, 60
Creative Computing 215
Crimson Crown 112
Curse of Ra 112
cursor control keys 12, 66
CZ-101 101, 102, 103, 135
CZ-230S 101
CZ-3000 101, 102, 103

D

DATA LIBRARIES 98
Data Manager 78
data storage 14-16
database fields 76
database files 76-7, 79
database functions 76-7
database management system 75-9
database templates 78
DB-One 78
debugging BASIC programs 148-50
Debug window, Logo 170
DEGAS 100, 123-5, 128, 131, 133, 160
Delay 167
DELETE 148
deleting files 59-60
Delphi 96, 97, 224
DESK menu 31, 40-48
desktop 24, 26, 31, 34, 60-2
Desktop Info 42
desktop publishing 212
DEVPAC ST 194
DFT 86, 215-6
Dialogue window, Logo 170
Diction 64
Digisound ST 135
Digital Equipment Corporation 87, 88
Digital Vision 133
DIM 159
direct file transfer 86
disk backup 216
disk capacity 54
disk copying 56-8
disk drive 15-16, 17, 20 ,22, 25, 35, 54, 208, 215
disk drive port 20
disk formatting 54-56
Disk Identifier 35, 49
disk protection 10
disks 25, 54-60
disks, write protection 24, 25
Donald's Playground 117
double-sided drives 15, 20, 56
Dow Jones 81, 94
downloading programs 85, 86, 87, 88, 95, 100
draft print mode 47
dragging icons 51-2
Dragon Group, The 121, 205
drill and practice 116
drop-down menus 28, 31-39
Duplicating Technologies 217
duration 166

E

Easy Draw 125-28, 160
EasyLink 91-2
Eclipse Engineering 117
Edit menu, BASIC 143, 144, 145
Edit menu, Logo 180
edit mode, BASIC 142-46
Edit window, BASIC 139, 146
Edit window, Logo 170, 180
educational applications 6, 116-18, 168-9
educational simulations 117
EIES 93
Electro Solitaire 110
electronic desktop 24, 26, 31, 34, 60-62
electronic file cabinet 75
electronic forum 97
Electronic Information Exchange System 93
electronic mail 88, 91-2
electronic networking 88, 92-3
electronic publishing 88, 93-4
electronic spreadsheets 70, 72-5
Elementary Atari ST, The 219
ELLIPSE 164-5
END 148
Ensoniq Mirage 135
Enable 167
Envelope 167
Epyx 115
error correction protocol 85, 86, 87, 88, 98
Escape (ESC) key 12
Express Mail 91

F

Fast/BASIC-M 197
Federal Express 91
FENCE 184
fields, database 76
file icons 29-31
file management system 75
FILE menu 31, 55-60, 62
files 50, 51-2, 58-60
files, database 76, 79
FILL 163
Fill options and patterns, Logo 177
filters, power-line 11
final print mode 47
Final Word 68
financial services 88, 94-5
Firebird 112
Fitler, William 221
Flight Simulator 114
folders 51-2, 60
footnoting and indexing 67
FOLLK 186
FoREM ST 91
Format 55
formatting disks 54-6
FOR...NEXT 156-7
Forth 205-6
Forth Dimensions 205
Forth Interest Group
FORTRAN 96, 200, 206
FORTRAN-77 206
forums 97
FORWARD 171-2
freeware 95
FTL 112
Full Box 49, 140
FULLSCREEN (FS) 173
FULLW 150
function keys 12, 13
function keys,BASIC 144-46

G

games of chance 106, 108
GEnie 92, 93, 96
GEM 23, 24, 31-34, 36, 40-62, 165, 194, 204
GEM-based word processors 65, 69
GEM Operating System Handbook 221
GEM Programmer's Reference 220
GEMdraw 40
GEMwrite 40
GOSUB...RETURN 157-8
GOTO 156-8
grammar checkers 64-5
Grammatik 64
Graphics and Sound 220
Graphic Artist, The 128-30
graphics, BASIC 160-66
graphics, bit-mapped 161
graphics, Logo 168-84
Graphics Display Window, Logo 170
Graphics Environment Managers 23, 24, 31-34, 36, 40-62, 165, 194, 204
Graphics Function Library 204
graphics settings, Logo 177-9
graphics software 119-34
GST ASM 194
GSTC Software Development Package 194, 205
Gumball Express 15

H

H & D Base 77, 100
H & D Tool Box 217
H & R Block 96
Haba Systems 16, 70, 105, 106
HabaWriter 70
Hacker 112
hacker 7
hard disk 15-16, 216
hard disk port 20
Harman, Thomas 193
Hayes 84, 85, 86
Hayes-compatible 84, 86, 91
Heilborn, John 219
Held, Gilbert 219
Help Calc ST 75
HELP key 13
Help Edit, BASIC 145
Henry's Fundamental BASIC 197
Hex 115
hexidecimal-decimal conversion 191
hexidecimal numbers 188, 189, 190
HIDETURTLE (HT) 172
high-level languages 195, 196-206
high resolution graphics 14
high resolution mode 33, 160
HippoBackgammon 107
HippoConcept 69, 71
HippFonts 69

HippoPixel 69
Hippo Ramdisk 217
Hippopotamus Software 69, 71, 107, 135, 217
HippoSpell 69
Hippovision Video Digitizer 133
HippoWord 69
Hisoft 194
Hitchhiker's Guide to the Galaxy 113
Holmes & Duckworth 205, 217
HOME 172
home applications 6, 104-6
HomePak 81
home position, turtle 171
Hudson, Tom 128
Hybrid Arts 135

I

IBM PC 13, 15, 17, 19, 40, 78, 79, 85, 86, 191, 198
icons 28-29, 51
idea processing 70-1
IF...THEN 155-6
immediate mode, BASIC 142
Index 164
indexing and footnoting 67
InfoGlobe 81
Information Line 49-50
information management 88, 89-90
information utilities 89, 96
informed user 7
INPUT 152, 154
Insoft 204
insert and delete modes 65
Install Application 36
Install Disk Drive 35-36
Install Printer 46-47
instruction sets 187-8
integrated software 79
Integral Solutions 81
interpreters 197
Introduction to Sound and Graphics on the Atari ST 220
Introduction to ST Logo 186
investment software 81
I/O ports 18
*I*S Talk* 81
Isgur Portfolio 81

J

Jacintosh 2
Jet 114
joystick 18, 22, 213-4
joystick port 18

K

Kermit 85, 88
Key Repeat Control 43
Key Repeat Speed 43
keyboard 12-13, 60-61
Keyboard Response 43
keywords, BASIC 146-7
Knight, Tim 220

L

Lattice C 205
Lawson, Barbara 193
LDW BASIC ST 197
LDW Software 197
Learning Forth 206
LEFT 172
LET 151, 152-3
LINEF 165
line numbers, BASIC 145-6, 148
line settings, Logo 178-9
LIST 145
List window 139, 140, 141
literal string 151
LOAD 147
logical conditioners 156
logical operators 188-9
Logik Corporation 21
Logo 4, 118, 168-86, 196, 200, 203, 205
Logo editing 180
Logo primitives 170, 172, 179
Logo procedures 179, 181
Logo User's Guide 220
loops 156-7
Lotus 1-2-3 6, 79
low resolution mode 33, 44, 160, 161

M

M-Disk RAM 217
machine language 5, 187, 191-3, 199

Macintosh 1, 16, 22, 38
macros 193
Macro Assembler 205
Magic Slate 120
mail-order suppliers 208-9
mainframes 47-8, 85, 87-8, 95-6
mainframe terminal emulation 47-8, 85, 87-8
Major Motion 114
Mark of the Unicorn 68, 87, 115
Mastering GEM 221
Mastering Forth 206
Mathmagics 117
Mean 18 115
medium resolution mode 33, 160, 161
Megamax 205
Megamax C 205
memory 14, 208
memory upgrade kits 209, 210
memory address 192, 195
menu bar 31, 40, 138, 143
menu-driven word processors 65
menus 28, 31-39, 55-60, 62
Meta Assembler 194
MichTron 86, 106, 110, 114, 125, 215, 217
MichTron Utilities 217
microprocessor 5, 16, 187-8, 193
Microsoft BASIC 196
Microsoft Word 68
Micro-W Distributing Company 136
MIDI 5, 18, 99, 101, 102, 103, 134, 135, 208
MIDI MAGIC 136
MIDI ports 18, 92, 135
Migraph 125
Miller Computer Products 217
Mind Mine 205
Mirage Concepts 77
Mi-Term 86
model expert 7
modem 84-6, 89, 91, 214
MODEM7 85
modem port 20, 23, 47
Modula-2 4, 99, 203-4
monitor 13-14, 18, 22, 26, 27, 208, 213
monitor connections 13
monitor port 18
Motorola 68000 chip 5, 16, 17, 187, 193
mouse 18, 22, 26, 28
Mouse Click Response 43
mouse pointer 26 60-61
mouse port 18
Move Bar 49, 50, 52
moving icons 51-2
moving windows 51-2
MT-Forth-83 205
music software 134-6
Music Studio 101, 102, 103, 134, 135
music synthesizers 101, 102, 103
multi-tasking 67, 205

N

National Logo Exchange 186
NEOchrome 99, 120-23, 131, 160
networking 92-3
NEW 147
New Folder 60
NewsNet 94
Norton Utilities 217
Note 166
Novation 90
null-modem cables 86
number systems 189-91
numeric keypad 12
numeric variable 152-3, 158

O

object code 191
Octave 165
Okidata 46, 212
Okimate 20 46, 119
one-dimensional array 159
opcodes 192
Open 62
opening files 51, 62
opening windows 28, 62, 141
OPENW 141
operation codes 192
operating system 23, 26, 194
Optimized Systems Software 203, 217
OPTIONS menu 31, 160
Osborne/McGraw Hill 219
Output window, BASIC 139, 141, 151
Overnight Mail 91
Owner's Guide, Atari ST 9, 25
Oxxi, Inc. 78

P

p-code, Pascal 199
page formatting and printing 66
Palette Control 44-45
PaperClip Elite 71
Papert, Dr. Seymour 168-9
parallel port 19, 47, 207
parallel printer 19, 23, 212
PARTI 93
Pawn, The 111
PC/InterComm 87
PC Magazine 81
Pascal 4, 5, 96, 195, 198-203
Pascal procedures 202-3
PCIRCLE 164
Peeks and Pokes 220
Pegammon 107
PELLIPSE 165
pen state, Logo 173
PENDOWN (PD) 173
PENERASE (PE) 173
Penguin Software 112
PENUP (PU) 172
Period 166
Personal Computing 22, 70
Personal Diskit 217
Personal Money Manager 106
Personal Pascal 203
Philon 197, 205
Piaget, Jean 168-9
pixel 33, 47, 160-1
plotters 212-3
POKE 165
pointer 26
port, cartridge 20-21
port, disk drive 20
port, hard disk 20
port, joystick 18
port, MIDI 18, 92, 135
port, modem 20, 23, 47
port, monitor 18
port, mouse 18
port, parallel 19, 23, 47, 207
port, printer 19, 23, 47
port, serial 20, 47
ports, I/O 18-21
power connector 18
power-line filters 11
powerstrips 10-11, 22
power supply 9, 10, 18, 22, 23
Power Systems 123
Powervision 123
Presenting the Atari ST 219
primitives, Logo 170, 172, 179
PRINT 141-2, 150, 151, 185
print modes 47
Print Screen 31, 35
printer 19, 23, 46-47, 208, 210-12
printer cable 19, 23, 47, 212
printer, color 46, 212
printer, daisy wheel 46, 211
printer, dot matrix 46, 210-11, 212
printer, laser 211-12
printer, parallel 19, 23, 212
printer port 19, 23, 47
printer ribbons 218
Printmaster 131
procedures, Logo 179-83
procedures, Pascal 202-3
Prochnow, Dave 220, 221
professional applications 72-82
program mode, BASIC 142
program user 6
programming, assembly language 5, 187, 193-4
programming, BASIC 4, 5, 36, 137-67
programming, C 4
programming, Logo 4
programming, machine language 5, 187, 191-3
programming, Modula-2
programming, Pascal 4, 5, 198, 200-3
Progressive Computer Applications 128
Pro FORTRAN-77 206
PRO Pascal 206
Prospero Software 206
public domain software 95, 99, 100, 223
publishing, electronic 88, 93-4

Q

QRS 135
Quantum Microsystems 78, 92
Quickview Systems 78

R

Radio Shack 10,11, 85, 86
Random Access Memory (RAM) 14, 191, 209
RAM disks 217

Read Only Memory (ROM) 14, 2, 191, 194
recreational computing 5, 106-115
recursion 183
Regent Base 78
Regent Spell 64
Regent Software 64, 69,78
Regent Word 69
renaming files 51
RENUM 148
REPEAT 174
resolution, high 33
resolution, low 33
resolution, medium 33
resolution, screen 32
RF modulator 13, 14
RGB monitor 13, 14
RIGHT 172
Rogue, The 115
Royal Software 74, 75
RS232 84
RS232 configuration 47
Rubber Stamp ST 131
Rubin, Charles 22
RUN 78
RUN 145, 147
RUN menu, BASIC 146
RUNning BASIC programs 146

S

Sampler, The 121
Sanders, William 219
SAVE 147
Save Desktop 31, 34-35
Sawusch, Mark 220
SC1224 monitor 13, 14, 160
SC1424 monitor 13
SCI Software Development 194
Scrabble 107
screen colors 44-45, 162
screen coordinates, Logo 176, 177-8
screen design 150-2
screen resolution modes 32, 161
screen pointer 26, 60-61
screen settings, Logo 179, 182-4
scroll bar 53-4, 140, 151
scroll box 53, 140, 151
scrolling windows 53-4
SD-2000 disk drive 15, 215
search and replace 66
selecting icons 28, 51, 61, 62
serial port 20, 47, 207
Set Preferences 31-2, 160
Set RS232 Configuration 47
Set Screen Resolution 32
SETBG 179
SETFILL 178
SETHEADING (SETH) 175
SETLINE 179
setting screen colors 44-45
Settings menu, Logo 173-80
set-up procedures 22-23
SETX 176
SETY 176
SF314 disk drive 15, 20, 56, 215
SF354 disk drive 15, 20, 56, 215
Shanner International 15, 215
Shape 166
shareware 95
Shivji, Shiras 3
SHOW 185
Show as Icons 37
Show as Text 37
SHOWTURTLE (ST) 172
Sierra On-Line 117
SIGATARI 97, 98
SIGs 92, 93, 96, 97, 134
sign on procedure 90
Simpson, Henry 221
simulations 107, 110-15, 116, 117
Singer, Matthew 87
Size Box 53, 140
SM124 monitor 160
Small, David 194, 215
Small, Sandy 194, 215
SofTech 199
Softlogik Corporation 110
Soft Spool 217
software, bulletin board system 91
software, database 75-9
software, educational 116-18
software, graphics 119-34
software, integrated 79
software, investment 81
software, music 134-6
software, public domain 95, 99, 100, 223-4
software, recreational 106-115
software, sources 223-4
software, spreadsheet 72-5
software, telecommunications 86-8
software, user-supported 95
Solutions: Wills 106
SOUND 166

sound and graphics 119-36
sound digitizer 135-6
sound synthesis 166
Source, The 6, 81, 89, 92, 93, 96
source codes 191-3
SPC 152
Special Interest Groups 92, 93, 96, 97, 134
spelling checkers 64
spreadsheets 70, 72-5
spreadsheet templates 75
ST Calc 74, 75
ST Coloring Book 121
ST Developer's Kit 194
ST Duplicator 217
ST-File 78
ST Internals 220
ST Language Disk 26, 120, 138, 168
ST-LOG 208
ST-NET 92
ST Planetarium 117
ST Programmer's Guide 220
ST Software Catalog 118, 194
ST Sound Digitizer 135
ST-TERM 87-8, 224
ST Tricks and Tips 220
START 208, 215, 221-2, 224
Starting Forth 205
start-up procedures 24-28
structured programming 198, 203
Style 164
string variable 152-4, 158, 160
Sublogic 114
subroutines, BASIC 157-8
subroutines, assembly language 193-4, 195
subroutines, Pascal 203
subscripted variables 158--60
Sundog Frozen Legacy 112
SuperCalc 17
Supra 16, 216
SwiftCalc 74
System Disk 24, 25, 26, 41
systems operators (SYSOPS) 90, 91, 97, 98

T

T-shirt transfers 218
TAB 152
TAB Books 220-1
tab, write-protect 24, 25
TDI Software 199, 203, 204, 206
Techstar, 194
telecommunications 6, 17, 81, 83-104, 214
Telenet 92
television 13
templates, database 78
templates, spreadsheet 75
Temple of Asphai 112
terminal emulation 47-8, 87, 88
The Home Computer Wars 2
The Motorola MC68000 Microprocessor Family 193
The Source 6, 81, 89, 92, 93, 94, 96
Thinking Forth 205
thinking skills and Logo 169
Time and Date 43
Time Bandit 114
timesharing 96
Timeworks 74, 78
Tomczyk, Michael 2
TOS 14, 24, 27, 36, 40-62, 81, 194
TOS System Disk 24, 25, 26, 41, 106, 138
trackballs 214
Tracy, Martin 206
tractor feed paper 47
Tramiel, Gary 2, 3
Tramiel, Jack 2, 3
Tramiel, Leonard 2, 3
Tramiel, Sam 2, 3
Tramiel Operating System 14, 24, 27, 36, 40-62, 81
Tramiel Technology Limited 3
Transylvania 112
trash can 37-38
TRACE 149-50
Trageser, James 221
TRON 149-50
Turbo Pascal 198-9
Turbo Tutor 198
turtle boundaries 184
turtle graphics 168-84
turtle heading 175
turtle position 173, 176
turtle settings 173-6
turtle state 173
tutorial 116, 117
two-dimensional arrays 159-60
Tymnet 92
TYPE 185
Typesetter ST 131
Typing Tutor 117

U

UCSD Pascal 199, 203
Understanding the Atari ST 220
Undo key 12
Uninet 92
Unison World 131
uploading programs 86, 87
Upper Reaches of Asphai 112
user-supported software 95
users group 99, 223-4

V

variable names 153
variable, numeric 152-3, 159
variable, string 152-4
variables 152-4, 155, 158-60, 183
VAX 87
VDI 165
VDISYS 165
video digitizer 133-4
video display 13, 213
video game controllers 213
VIEW menu 31, 37
VIP Light 81
VIP Professional 80
VIP Technologies 80
Virtual Display Interface 165
Voice 166
Volume 166
VT52 Emulator 47-8
VT52 terminal 87, 88
VT100 terminal 87
VT102 terminal 87

W

Warner Communications 3
WAVE 166
Weber Systems Incorporated 219
West Coast Computer Faire 16
WHILE DO 202
WIDTH 150
WINDOW 184
windows 28, 48-54, 62
windows, BASIC 138-41
windows, Logo 170
Winnie the Pooh 113
Wirth, Niklaus 198, 203, 204
Word for Word 107-8
Word Invaders 117
word processing 63-71
word processors, command-driven 65
word processors, features 65-7
word processors, menu-driven 65
word processors, selection 65-7
Wordstar 17
World, The 121
WRAP 184
write-protect tab 24, 25

X

Xanth Computer Systems 218
XLent Software 131
x,y coordinates, BASIC 161-3
x,y coordinates, Logo 176-7
XMODEM 85, 86, 87, 88, 98

Y

YM 2149 chip 17
Yamaha 179
Yamaha DX 135
Yamaha TX 135
Young People's Logo Association 186

Z

Z80A chip 16, 17
Zap Mail 91
Zoomracks 78
Zork 113
Zork III 113

WITHDRAWN

CAMERON VILLAGE APR 1 4 1987

001.64
ATARI
WILLIS
034200

THE ESSENTIAL ATARI 520ST AND 1040ST
USER'S GUIDE

WAKE COUNTY PUBLIC LIBRARIES
RALEIGH, N.C.